

Universidad Carlos III de Madrid

Escuela Politécnica Superior



Grado en Ingeniería Informática

Trabajo Fin de Grado

**“Búsqueda de relaciones en grafos extraídos de
redes sociales”**

Autor: José Ángel Cuadrado Mingo

Tutor: José Arturo Mora Soto

Junio de 2012

Agradecimientos

En primer lugar, dar las gracias a Jessica por ofrecerme la posibilidad de desarrollar este proyecto, y a Arturo por continuar la labor que empezó Jessica.

Quiero dar también las gracias a todos mis compañeros del Laboratorio (Óscar, Roberto, Jaime, Christian, Lidia, Iván, Alberto, Alicia, Ernesto, Álex, Noelia, Álvaro), ya que sin ellos, no habría aprendido tantas cosas como las que se ahora.

También agradezco a los chicos de Dele (Mario, Javi, Irene, Bailón, Ortega, Fernando, Adela, Miguel, Luis) el haber estado ahí siempre, compartiendo buenos momentos y ayudando a todos los estudiantes con sus dudas y problemas. También doy las gracias en especial a Adri, por todos sus consejos informáticos, además de lo mencionado anteriormente.

Quiero dar las gracias también a todos mis chicos de informática, por haber pasado tantos momentos juntos: a “Nerelu” y Rodry, por estar siempre ahí y poder contar con ellos; a Baldo, por ser como es y poder disfrutar de los buenos y malos momentos con él; a Rober, Sole y Javi, por haber pasado tanto tiempo juntos haciendo práctica tras práctica; al “Canario”, por haberme soportado con tantas dudas y haberme ayudado con las prácticas; a “Funny”, Marta y Bea, por hacer el periodo en la universidad más ameno; a Borja, Mery, Aponte, Óscar y Omar, por jugar al baloncesto juntos; a Juanfran, Casillas, Christian, Aitor y Elizaga, por esos momentos y partidas al mus al final de los exámenes; a Tamara, por mandar callar a la clase y reivindicar sus derechos; a Pablo, Miguel, Javi, Ana y Jessica, por compartir las comidas en el comedor y ese genial viaje a Amsterdam; a Joel y Adri, por ser unos trabajadores incansables; a Gabri y Álvaro, por enseñarme todos los trucos existentes para todo; y finalmente, mencionar también a Camelia, Jenny, Raquel, Edu, Vicen, Cristian, Álvaro, Godoy, José Luis, Yolanda, Patri, Tremps e Iván; muchas gracias a todos.

Gracias también para Lau, Noe, Ana y Deborah, por estar también ahí cuando lo necesitaba.

Doy las gracias a toda mi familia, y en especial a mis padres, por ser tan maravillosos como son y apoyarme siempre en todo y con todo, e intentar rectificarme cuando creen que me equivoco.

Y antes de acabar, dar en especial las gracias a Nere, mi amor, por estar siempre a mi lado constantemente, apoyándome en todo momento para poder realizarlo –a pesar de no gustarle el tema del proyecto-, e intentando ayudarme siempre que podía con todo y animándome cuando estaba más decaído. Muchísimas gracias por todo, enana.

Y en definitiva, dar gracias a todo aquel que ha pasado por mi vida, ya que sin ellos, no sería quien soy ahora.

Gracias.

Índice

1.	Introducción.....	11
1.1.	Contexto.....	11
1.2.	Objetivos.....	12
1.3.	Estructura del documento	13
2.	Estado del Arte	14
2.1.	¿Qué es una red social?.....	15
2.2.	Historia de las redes sociales	17
2.3.	Redes sociales más importantes en la actualidad.....	18
2.4.	Privacidad en redes sociales	20
2.4.1.	Tuenti.....	20
2.4.2.	Twitter	22
2.4.3.	Facebook.....	23
2.4.4.	Conclusiones.....	25
2.5.	Aplicaciones en Facebook	26
2.5.1.	Desarrollo de aplicaciones en Facebook	29
2.5.2.	Ejemplo de creación de una aplicación en Facebook	34
2.6.	Librerías externas.....	39
2.7.	Conclusiones.....	40
3.	Estudio de Viabilidad del Sistema.....	41
3.1.	Introducción	41
3.2.	Alcance del sistema	42
3.2.1.	Identificación de los interesados en el sistema (stakeholders)	43
3.2.2.	Planificación	43
3.2.3.	Presupuesto.....	45
3.3.	Situación actual.....	46
3.4.	Definición de requisitos de usuario	47
3.4.1.	Requisitos de capacidad.....	48
3.4.2.	Requisitos de restricción.....	51
3.5.	Estudio de alternativas de solución.....	54
4.	Análisis del sistema	55
4.1.	Introducción	55
4.2.	Definición del sistema	56
4.2.1.	Determinación del alcance del sistema.....	56
4.2.2.	Identificación del entorno tecnológico	56
4.2.3.	Especificación de estándares y normas	57

4.2.4.	Identificación de los usuarios	58
4.2.5.	Estudio de la seguridad requerida en el proceso de análisis.....	59
4.3.	Establecimiento de requisitos software.....	60
4.3.1.	Obtención de requisitos software	60
4.3.2.	Especificación de casos de uso.....	73
4.4.	Análisis de la estructura del sistema	81
4.5.	Definición de las interfaces de usuario	83
4.5.1.	Especificación de principios generales de la interfaz.....	83
4.5.2.	Identificación de perfiles y diálogos.....	83
4.5.3.	Especificación de formatos individuales de la interfaz de la pantalla.....	83
4.6.	Análisis de consistencia y especificación de requisitos.....	85
4.6.1.	Verificación y análisis de consistencia entre modelos	85
4.6.2.	Validación de los modelos.....	87
5.	Diseño del sistema.....	88
5.1.	Introducción	88
5.2.	Definición de la arquitectura del sistema.....	89
5.2.1.	Definición de niveles de arquitectura	89
5.2.2.	Especificación de estándares y normas de diseño y construcción.....	89
5.2.3.	Especificación del entorno tecnológico	90
5.2.4.	Especificación de requisitos de seguridad y operación	90
5.2.5.	Estudio de la seguridad requerida en el proceso de diseño del sistema	90
5.3.	Diseño de la arquitectura de soporte	91
5.4.	Diseño físico de datos	93
5.5.	Establecimiento de requisitos de implantación.....	95
5.5.1.	Especificación de requisitos de implantación.....	95
6.	Implementación del sistema	96
6.1.	Interfaz de la aplicación.....	96
6.2.	Algoritmo de búsqueda.....	99
6.2.1.	Primer nivel de profundidad	103
6.2.2.	Segundo nivel de profundidad	103
6.3.	Muestra de resultados	104
7.	Validación y verificación del software	106
7.1.	Introducción	106
7.2.	Plan de pruebas	107
7.2.1.	Elementos de prueba.....	107
7.2.2.	Necesidades del entorno	108
7.2.3.	Criterio de aceptación de las pruebas	108

7.3.	Procedimientos de prueba	109
7.4.	Análisis de consistencia de pruebas	112
7.5.	Informe de pruebas	113
7.6.	Pruebas de usuario	122
8.	Conclusiones.....	124
8.1.	Líneas futuras.....	125
8.2.	Consideraciones éticas	126
	Glosario	127
	Bibliografía.....	129
	Anexo A: Planificación	133
	Anexo B: Presupuesto	134
	Anexo C: Datos de la encuesta.....	135

Índice de figuras

Imagen 1. Mapa del mundo de las redes sociales.....	18
Imagen 2. Amigos solicitando ayuda en el juego 'The Sims Social'.	27
Imagen 3. Personaje interactuando con un vecino.	27
Imagen 4. Aplicación que realiza preguntas sobre los amigos de un usuario.	28
Imagen 5. Grafo de amistades.	29
Imagen 6. JavaScript Test Console	32
Imagen 7. Graph API Explorer.....	33
Imagen 8. Creación de una nueva aplicación	34
Imagen 9. Configuración básica de la aplicación	34
Imagen 10. Integración de la aplicación en Facebook.....	35
Imagen 11. Ejemplo del funcionamiento de la aplicación de ejemplo.....	38
Imagen 12. Grafo generado con Protovis	39
Imagen 13. Diagrama de Gantt de la planificación	44
Imagen 14. Casos de uso del sistema	74
Imagen 15. Estructura de la aplicación.....	81
Imagen 16. Boceto de la interfaz gráfica.....	84
Imagen 17. Definición de niveles de arquitectura	89
Imagen 18. Componentes del sistema	91
Imagen 19. Ejemplo de datos en Facebook	93
Imagen 20. Comunicación de datos.....	94
Imagen 21. Confirmación de permisos de la aplicación.....	96
Imagen 22. Interfaz gráfica inicial de la aplicación.....	97
Imagen 23. Sección de ayuda	97
Imagen 24. Aplicación en el proceso de búsqueda.....	98
Imagen 25. Algoritmo de búsqueda (vista apaisada).....	99
Imagen 26. Llamadas asíncronas de Facebook	100
Imagen 27: Grafo resultante de una búsqueda.....	104
Imagen 28. Mejores resultados obtenidos de una búsqueda.....	105
Imagen 29. Resultado final de una búsqueda	105
Imagen 30. Resultado de la prueba 1	114
Imagen 31. Resultado de la prueba 2.....	114
Imagen 32. Resultado de la prueba 3.....	115
Imagen 33. Resultado de la prueba 4.....	115
Imagen 34. Resultado de la prueba 5.....	116
Imagen 35. Resultado de la prueba 6.....	116
Imagen 36. Resultado de la prueba 7.....	117
Imagen 37. Resultado para la prueba 8.....	117
Imagen 38. Resultado para la prueba 9.....	118
Imagen 39. Resultado para la prueba 10 (cerca)	118
Imagen 40. Resultado para la prueba 10 (lejos)	118
Imagen 41. Resultado para la prueba 11.....	119
Imagen 42. Resultado para la prueba 12 (zona original).....	119
Imagen 43. Resultado para la prueba 12 (zona nueva).....	119
Imagen 44. Resultado para la prueba 13 (nodos originales).....	120
Imagen 45. Resultado para la prueba 13 (nodo arrastrándose).....	120
Imagen 46. Resultado para la prueba 14.....	120
Imagen 47. Resultado para la prueba 15 (sin limpiar).....	121
Imagen 48. Resultado para la prueba 15 (limpiado).....	121

Imagen 49. Planificación (vista apaisada)	133
Imagen 50. Presupuesto	134
Imagen 51. Porcentaje de “Sexo” en la encuesta.....	135
Imagen 52. Porcentaje de “Edad” en la encuesta	135
Imagen 53. Porcentaje de “Ocupación profesional” en la encuesta	135

Índice de tablas

Tabla 1. Ejemplo de llamada al método FB.api	30
Tabla 2. Ejemplo de llamada al método FB.ui	31
Tabla 3. Código de ejemplo finalizado.....	37
Tabla 4. Planificación	44
Tabla 5. Plantilla de los requisitos de usuario	47
Tabla 6. RU-C-01	48
Tabla 7. RU-C-02	48
Tabla 8. RU-C-03	49
Tabla 9. RU-C-04	49
Tabla 10. RU-C-05	49
Tabla 11. RU-C-06	50
Tabla 12. RU-C-07	50
Tabla 13. RU-C-08	50
Tabla 14. RU-C-09	51
Tabla 15. RU-C-10	51
Tabla 16. RU-R-01	51
Tabla 17. RU-R-02	52
Tabla 18. RU-R-03	52
Tabla 19. RU-R-04	52
Tabla 20. RU-R-05	53
Tabla 21. RU-R-06	53
Tabla 22. Plantilla de los requisitos de software	60
Tabla 23. RS-F-01	62
Tabla 24. RS-F-02	62
Tabla 25. RS-F-03	62
Tabla 26. RS-F-04	63
Tabla 27. RS-F-05	64
Tabla 28. RS-F-06	64
Tabla 29. RS-F-07	65
Tabla 30. RS-F-08	65
Tabla 31. RS-F-09	66
Tabla 32. RS-F-10	66
Tabla 33. RS-F-11	67
Tabla 34. RS-F-12	67
Tabla 35. RS-F-13	67
Tabla 36. RS-F-14	68
Tabla 37. RS-F-15	68
Tabla 38. RS-F-16	68
Tabla 39. RS-F-17	69
Tabla 40. RS-F-18	69
Tabla 41. RS-I-01	69
Tabla 42. RS-I-02	70
Tabla 43. RS-I-03	70
Tabla 44. RS-I-04	70
Tabla 45. RS-I-05	71
Tabla 46. RS-R-01	71
Tabla 47. RS-R-02	71
Tabla 48. RS-S-01	72

Tabla 49. RS-S-02	72
Tabla 50. RS-C-01	73
Tabla 51. Plantilla de los casos de uso	73
Tabla 52. CU-01	75
Tabla 53. CU-02	75
Tabla 54. CU-03	76
Tabla 55. CU-04	76
Tabla 56. CU-05	77
Tabla 57. CU-06	78
Tabla 58. CU-07	79
Tabla 59. CU-08	79
Tabla 60. CU-09	80
Tabla 61. CU-10	80
Tabla 62. Matriz de trazabilidad RS-RU	86
Tabla 63. Matriz de trazabilidad RS-CU	87
Tabla 64. Plantilla de componentes	91
Tabla 65. C-001	92
Tabla 66. C-002	92
Tabla 67. RI-01	95
Tabla 68. RI-02	95
Tabla 69. Búsqueda en el campo "estado civil".	102
Tabla 70. Plantilla de especificaciones de pruebas.....	109
Tabla 71. PR-01	109
Tabla 72. PR-02	109
Tabla 73. PR-03	109
Tabla 74. PR-04	110
Tabla 75. PR-05	110
Tabla 76. PR-06	110
Tabla 77. PR-07	110
Tabla 78. PR-08	110
Tabla 79. PR-09	110
Tabla 80. PR-10	111
Tabla 81. PR-11	111
Tabla 82. PR-12	111
Tabla 83. PR-13	111
Tabla 84. PR-14	111
Tabla 85. PR-15	111
Tabla 86. Matriz de trazabilidad PR-RC	112
Tabla 87. Plantilla de informe de pruebas	113
Tabla 88. R-PR-01	114
Tabla 89. R-PR-02	114
Tabla 90. R-PR-03	115
Tabla 91. R-PR-04	115
Tabla 92. R-PR-05	116
Tabla 93. R-PR-06	116
Tabla 94. R-PR-07	117
Tabla 95. R-PR-08	117
Tabla 96. R-PR-09	118
Tabla 97. R-PR-10	118
Tabla 98. R-PR-11	119

Tabla 99. R-PR-12.....	119
Tabla 100. R-PR-13.....	120
Tabla 101. R-PR-14.....	120
Tabla 102. R-PR-15.....	121
Tabla 103. Pregunta 1 de la encuesta	136
Tabla 104. Pregunta 2 de la encuesta	137
Tabla 105. Pregunta 3 de la encuesta	138
Tabla 106. Pregunta 4 de la encuesta	140
Tabla 107. Pregunta 5 de la encuesta	142

1. Introducción

El presente documento técnico que aquí se detalla describe el desarrollo de una aplicación software dedicada a realizar búsquedas de relaciones. Estas relaciones, a día de hoy, se encuentran visibles a todo el mundo desde las redes sociales, mundo en el que se hará hincapié para descubrir todas sus posibilidades.

1.1. Contexto

Desde hace algunos años, se ha venido dando un cambio en la forma de utilizar Internet por los usuarios. A día de hoy, existe una mayor interactividad entre dichos usuarios con los sitios web que consultan, pudiendo comentar y opinar lo que allí se decía, y poder compartir esa información con otras personas.

En esta llamada Web 2.0, cabe destacar el efecto causado por las redes sociales, sitios web en los que la gente se registra para estar en contacto con otros usuarios y poder compartir desde cualquier información hasta ver las fotos de una noche de fiesta, pasando por encontrar a viejas amistades del colegio y crear eventos para quedar con la gente.

Con la popularidad de estas redes, cada día más gente se está apuntando a estos sitios web, haciendo que la cantidad de usuarios crezca exponencialmente, y cambie hasta los propios modelos de negocio de las empresas, que se orientan a darse a conocer por Internet, y en concreto, a publicitarse en las redes sociales.

En este ámbito, las empresas pueden llegar a realizar grandes estudios para conocer mejor que es lo que quieren los usuarios, o saber a qué tipo de usuarios puede orientarse su producto. Es por ello que buscan información sin parar en estas grandes redes que se están tejiendo en Internet para extraer todas las relaciones posibles que ayuden a vender dicho producto.

Sin embargo, no sólo las empresas tienen que buscar información en estas redes, si no que cualquier usuario puede estar interesado en encontrar a alguna persona que esté familiarizada con algún aspecto en concreto que necesite conocer.

Aprovechando que cada día existen más aplicaciones relacionadas con las redes sociales, basadas en la propia facilidad que dan éstas para poder crearlas, se estudia la posibilidad de crear una aplicación que permita buscar relaciones entre las diferentes personas de una red social con una información deseada, mostrando los resultados de una forma vistosa a través de un grafo.

1.2. Objetivos

Los objetivos que se plantean para este proyecto son los que se detallan a continuación:

- Desarrollar una aplicación software basada en un buscador de relaciones entre diferentes usuarios de una red social a través de la información almacenada en éstos y mostrando los resultados a través de un grafo.
- Realizar un estudio sobre las diferentes posibilidades que tienen las redes sociales para crear aplicaciones en ellas y obtener información de los usuarios a través de sus bases de datos.
- Diseñar un algoritmo que permita realizar una búsqueda y obtener los resultados deseados para la información que precisa el usuario.
- Crear una interfaz usable e intuitiva para los usuarios finales de la aplicación que permita mejorar su experiencia con la aplicación.
- Obtener información del usuario final para mejorar la herramienta en función a las diferentes opiniones obtenidas y mejorar la aplicación en el futuro.
- Seguir diferentes metodologías y estándares para la realización del proyecto, así como cumplir con los necesarios marcos legales para desarrollar la aplicación.
- Respetar la planificación y marco económico establecido para cumplir los plazos de entrega del proyecto.
- Poner en práctica los diferentes conocimientos aprendidos durante la carrera, mejorando las técnicas aprendidas y obteniendo nuevos aprendizajes sobre la materia.
- Proporcionar la información necesaria para mantener y ampliar la aplicación desarrollada.

1.3. Estructura del documento

Para desarrollar este documento, se ha seguido la estructura que se expone a continuación:

- **Introducción:** En él se explica el contexto en el que se sitúa el desarrollo del proyecto, así como los objetivos que se pretenden conseguir y la estructura del documento.
- **Estado del Arte:** En él se describe la situación actual previa al desarrollo del proyecto, analizando qué es una red social y cuáles son las principales redes que existen, viendo el marco legal en el que se sitúan éstas en cuanto a la privacidad, e investigando las diferentes formas de desarrollo de la aplicación.
- **Estudio de Viabilidad del Sistema:** Se describe el alcance que debe tener el sistema, en el que se incluye una planificación y un presupuesto, la situación actual en la que se encuentra y los diferentes requisitos obtenidos del cliente.
- **Análisis del Sistema:** En él se describen los diferentes requisitos software necesarios para desarrollar la aplicación, así como los casos de uso que pueda tener, la estructura, y la idea de la interfaz que debe tener.
- **Diseño del Sistema:** En él se definirá la arquitectura general que debe poseer el sistema, así como el diseño físico de los datos y su comunicación, y los requisitos de implantación necesarios para el sistema.
- **Implantación del Sistema:** En él se explicará las diferentes actuaciones para desarrollar el sistema, tales como la interfaz gráfica, el algoritmo de búsqueda y la muestra de los resultados obtenidos.
- **Validación y verificación del software:** Se realizará un plan de pruebas para la aplicación desarrollada, se mostrarán los resultados de este plan, y se realizarán encuestas a usuarios finales de la aplicación.
- **Conclusiones:** En él se resume las conclusiones tomadas durante la realización del proyecto, y presenta las posibles mejoras software que puede tener el sistema.
- **Anexos:** Incluyen información extra a la proporcionada en el documento para completar cualquier aspecto relevante de éste.

2. Estado del Arte

De un tiempo a esta parte, y gracias a la revolución de Internet en la sociedad, las personas han podido comunicarse de diferentes formas: foros, correos electrónicos, a través de servicios de mensajería instantánea, etc. Todos estos servicios han permitido la comunicación casi inmediata entre diferentes puntos del planeta que están muy alejados entre sí.

Con las nuevas tecnologías, se ha avanzado en el desarrollo de todo tipo de aplicaciones, y sobre todo, en el desarrollo de las comunicaciones a través de Internet. En este último aspecto, se han desarrollado desde múltiples navegadores hasta diferentes aplicaciones en las páginas web y nuevos servicios dentro de éstas. Esta nueva revolución tecnológica dentro de Internet es la llamada “Web 2.0” (Shelly & Frydenberg, 2010), en la cual se ubican diferentes servicios web de última generación como las wikis y las redes sociales.

Gracias a este último servicio que se ha mencionado, las redes sociales, las personas pueden, desde encontrar a un viejo compañero del colegio, hasta conocer todas las últimas novedades de un amigo viendo sus fotografías colgadas en la web, todo ello gracias a la red que se teje mediante la conexión de una persona con otra a través de un vínculo de amistad, profesión o gustos parecidos.

Para poder utilizar toda esta red que se ha tejido mediante las diferentes conexiones, hay que tener muy en cuenta las diferentes formas de buscar a una persona. En función de las características de la red social se puede buscar por su nombre y/o apodo, su edad, su ubicación, y por las redes a las que está unido (universidad, empresa, etc.). Además, todos los resultados obtenidos pueden incluso ser filtrados en función de si se quiere que se muestren sólo a aquellas personas que son amigos de un usuario, a las personas que se encuentren en un segundo grado de amistad (amigos de amigos), o a todas las personas de la red social. Sin embargo, este último aspecto depende del nivel de privacidad que tenga un usuario en su cuenta.

La privacidad de las personas es un aspecto muy importante a vigilar en las redes sociales. Si un usuario quiere publicar cierto contenido en Internet, debe tener cuidado con qué información o contenidos sube a la web, ya que a partir de ese momento, y teniendo en cuenta que Internet es público para todo el mundo, toda esta información publicada podrá ser vista por cualquier persona.

A continuación, se mostrará más información acerca de las redes sociales, conociendo en mayor profundidad en qué consiste este servicio, su historia y las redes sociales más importantes en la actualidad.

2.1. ¿Qué es una red social?

Una red social (Wikipedia, 2011) es una estructura social compuesta por personas que se conectan a su vez con otras, creando así una inmensa red de personas que comparten algún parentesco (amistad, familiar, compañeros de trabajo, etc.), algún gusto (musical, deportivo, social, etc.) o algún tipo de conocimiento.

Para poder entrar en una red social, lo primero que debe hacer el usuario es registrarse en dicha red. Existen diferentes tipos de registros, en la que una persona puede registrarse libremente, o debe ser invitada por otra que ya esté dentro de la red social. Para registrarse, el nuevo usuario deberá introducir una serie de datos personales en función de las características de esa red.

Una vez registrado, el usuario podrá crear su lista de amigos, añadiendo a gente conocida que ya esté en la red social, y unirse a las diferentes redes existentes (empresas, colegios, universidades, etc.). Además, el usuario podrá conversar con el resto de usuarios, compartir fotos y vídeos, e incluso participar en diferentes minijuegos, en los que hay contenida una alta carga social, pudiendo ayudar a un amigo para que vaya mejor en el juego, o competir contra él.

Se puede decir que la principal característica de las redes sociales es el poder contactar con otras personas a las que se haya agregado como “amigo”. De esta manera, se forma una gran red de personas unidas a otras por diferentes vínculos, pudiendo contactar con más personas que estén unidas a tus amigos. Esta característica tiene su base en la llamada “*Teoría de los seis grados de separación*” (Wikipedia, 2011).

Esta teoría, incluida en un estudio llamado “*Chains*” y realizado en 1930 por el escritor húngaro Frigyes Karinthy, decía que cualquier persona del mundo, estuviese lo lejos que estuviese, estaba conectado con cualquier persona a través de unos amigos o conocidos intermediarios, y postulaba que ese número de amigos o conocidos intermediarios no era superior a cinco (conectando a ambas con sólo seis enlaces).

El crecimiento exponencial de las personas conocidas en función de las ya existentes en la cadena con muy pocos enlaces entre unas y otras permite abarcar toda la población del planeta. Este hecho es la base de esta teoría.

Según lo que se acaba de describir, se puede exponer un ejemplo sencillo. Si una persona tiene 100 amigos, y cada uno de estos amigos, tiene otros 100 amigos, ésta persona puede llegar a tener contacto con 10.000 personas simplemente teniendo a un amigo de intermediario.

Estas 10.000 personas –que en verdad serían algunos menos, puesto que siempre hay personas que tienen amistades en común– serían contactos de segundo nivel: éstas son las típicas personas que salen en las conversaciones con los amigos acerca de lo sucedido a otra o de que necesita algo que la otra persona se lo puede proporcionar.

Si se sigue haciendo crecer esa pequeña red de segundo nivel que se acaba de formar, nos damos cuenta de que el número obtenido es enorme, hasta alcanzar, siguiendo la misma línea de amigos por persona, la cifra de 1.000.000.000.000 de personas en un sexto nivel, cifra que supera ampliamente la cantidad de personas existentes en el planeta, aún si se le resta la parte correspondiente a amigos comunes.

Lógicamente, cuanto más enlaces haya que saltar, más lejos será la distancia entre los nodos a conectar y más difícil será establecer la conexión. Gracias a Internet esta barrera ha sido eliminada.

2.2. Historia de las redes sociales

El inicio de las redes sociales (Mújica, 2010), se data en torno a 1995. En esa época, algunas páginas de Internet incluyeron la tecnología suficiente como para poder comentar en esos sitios web o crear listas de amigos, entre otras muchas posibilidades.

Ya en 1997 es cuando se hace referencia de la primera red social como tal, en la que se podían crear perfiles de usuario y listas de amigos de los usuarios que pertenecieran a dicha red social: SixDegrees.com.

Entre los años 1997 y 2001 se llevó a cabo una total revolución de las tecnologías web, que eran muy básicas por aquel entonces. En este último año, 2001, se creó la red social Ryze.com, que como gran novedad, estaba orientada al entorno empresarial, dándole a este tipo de webs una oportunidad de negocio dentro del mercado de la red.

En esta evolución de las webs, nació otra red social –Friendster- dedicada a concertar citas a través de la red. Sin embargo, no fue hasta el año 2003 y 2004 cuando se crearon dos de las principales redes sociales que aún existen hoy en día: MySpace y Facebook.

El fenómeno Facebook dio pie a la consolidación y desarrollo de numerosas redes sociales. Durante los años siguientes, se seguirían creando nuevas redes sociales en diferentes puntos del planeta. Cabe así destacar algunas como Orkut –en Brasil e India-, Mixi –Japón-, LunaStorm –Suecia-, Hi5 –Latinoamérica-, Bebo –Gran Bretaña y Oceanía-, QQ –China- y Tuenti –España-.

Tal fue la acogida de las redes sociales, que a día de hoy, la gran mayoría de las marcas de telefonía móvil permiten la posibilidad de conectarte a diferentes redes sociales desde cualquier lugar para poder compartir diferentes contenidos y/o información que se quiera con los amigos.

Pero, ¿qué han significado las redes sociales para los seres humanos? La popularidad de las redes sociales responde al deseo de las personas de relacionarse con su entorno. Sus ganas de contactar con otras son, en definitiva, modernas manifestaciones de esa tendencia innata de todos los seres humanos a ir más allá de sí mismos.

2.3. Redes sociales más importantes en la actualidad

En la actualidad, existen cientos de redes sociales en todo el mundo. Muchas de ellas son creadas para operar en un país o zona en concreto, tal y como se ha visto en el punto “2.2 Historia de las redes sociales”, mientras que otras están especializadas en diferentes ámbitos, como por ejemplo LinkedIn en el laboral.

En la imagen 1 (Ria Novosti, 2011), se observa un mapamundi con diferentes informaciones: por un lado, un mapa coloreado que muestra las redes dominantes en cada país; por otro lado, las redes sociales más populares en determinados países; y por un tercer lado, las redes sociales con más cantidad de usuarios registrados.

Tal y como se puede ver en dicha imagen, la red social más extendida en todo el mundo es Facebook, siendo la principal red social en una gran cantidad de países, entre los que se cuentan todos los del continente americano, excepto Brasil, Australia, India, el sudeste asiático, diferentes países africanos, y prácticamente todo Europa.

Además de ser la red social más extendida, Facebook también es la red social con mayor cantidad de usuarios en todo el mundo, seguida de cerca por la red social china QZone; sin embargo, a diferencia de la primera, todos sus usuarios pertenecen a China, al contrario que Facebook que los tiene repartidos por todo el mundo.

The world map of social networks

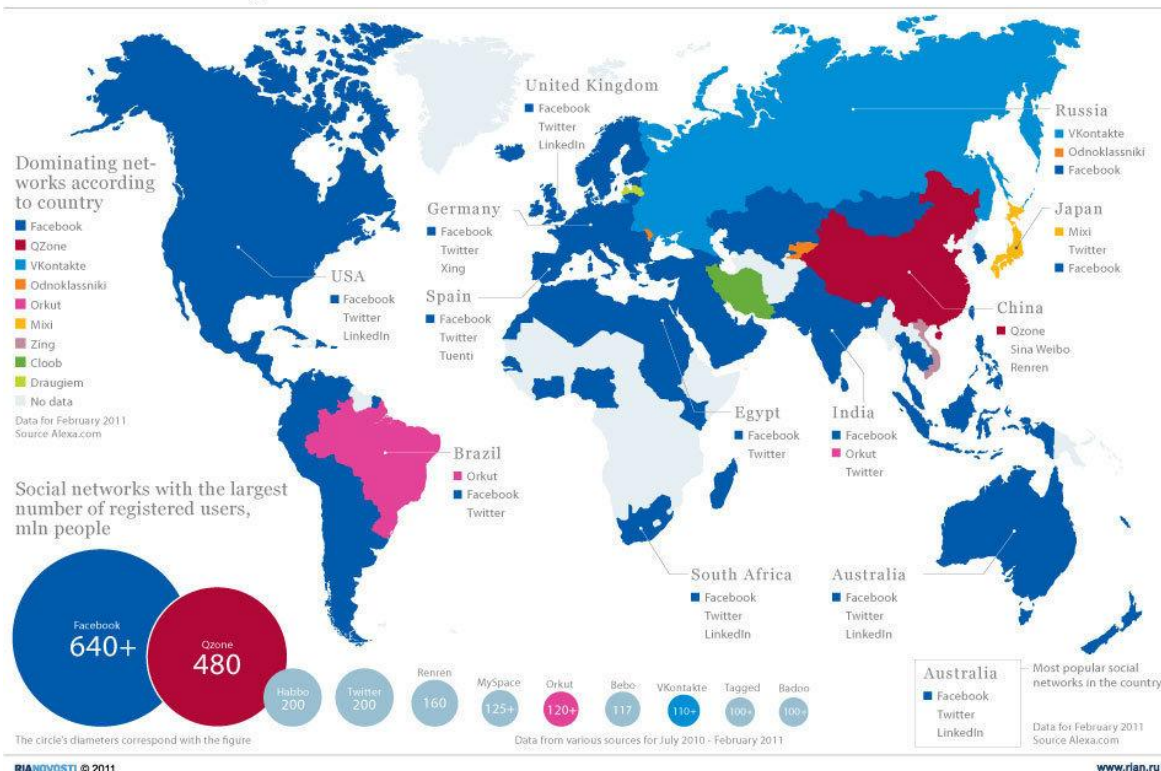


Imagen 1. Mapa del mundo de las redes sociales

Junto a Facebook, aparecen otras redes sociales bastante expandidas como son Twitter y LinkedIn. Sin embargo, en la fecha en la que se toman los datos para hacer la imagen 1 (febrero de 2011 con datos de la web “alexa.com”), aún no se había creado la red social de Google: Google Plus.

Google Plus nació a finales de junio de 2011 con la ambición de competir contra Facebook, que en ese momento, alcanzaba los 750 millones de usuarios registrados (Miller, 2011). Gracias a la integración de esta red social con otros servicios de Google, tales como su servicio de correo electrónico GMail, la cantidad de usuarios registrados se ha disparado, hasta superar los 90 millones de usuarios registrados en únicamente 7 meses (Romero, 2012). No obstante, muchos de estos usuarios están registrados en la red social únicamente por tener creado el perfil en otros servicios de Google, por lo que la cantidad real de usuarios de dicha red social es inferior. No obstante, su rápido crecimiento la convierte en una red social a tener en cuenta en el futuro, siguiendo de cerca su evolución.

2.4. Privacidad en redes sociales

Las redes sociales son una herramienta reciente que cuentan con una gran carga social y que manejan una enorme cantidad de datos privados proporcionados por los diferentes usuarios que se registran en ellas. Debido a ello, es obligatorio que todas las webs que manejen datos personales de un usuario tengan una política de privacidad –que debe ser aceptada por el usuario al registrarse- para controlar la utilización de esos datos.

Según la Real Academia Española, la privacidad (Real Academia Española, 2001) es el “ámbito de la vida privada que se tiene derecho a proteger de cualquier intromisión”. Según esto, cualquier dato o información de nuestra vida privada, debe ser protegida para que no sea conocida por cualquier otra persona que tenga fines lucrativos y/o vandálicos.

Además de lo dispuesto en el artículo 18 de la Constitución española de 1978 (Noticias Jurídicas, 1978), la principal ley española que regula la privacidad es la “Ley 15/99 Orgánica de Protección de Datos de carácter personal” (Noticias Jurídicas, 1999). Con estas normativas se pretende proteger el tratamiento de datos personales, especialmente lo concerniente a su honor, su intimidad y su privacidad, tanto personal como familiar.

En lo referente a Internet, los datos privados son iguales a cualquier otro, por lo que es necesario tener cuidado con dicha información, ya que Internet es de todo el mundo, y al no ser controlado por nadie concreto, cualquier persona puede acceder a cualquier contenido que se encuentre en él, y por tanto, perder la privacidad sobre esos datos.

En los siguientes puntos, se hablará de las redes sociales más importantes en España según la imagen 1, es decir, se estudiará la privacidad de la información utilizada y el desarrollo de aplicaciones sobre Tuenti, Twitter y Facebook.

2.4.1. Tuenti

La red social española fue creada a finales del año 2006 (Simón, 2008) con el fin de que ésta estuviese dirigida a jóvenes universitarios, aunque finalmente se dejó que cualquier persona pudiera registrarse en ella siempre y cuando otro usuario de la red social le invitase a ello.

Este último aspecto permite dar una mayor privacidad a sus usuarios respecto a otras redes sociales. Es por ello, que los usuarios tienen más confianza a la hora de subir sus datos privados, ya que si ellos lo configuran como tal, dicha información sólo podrá ser vista por sus amistades.

Un usuario de Tuenti puede subir a esta red social una gran cantidad de información. Para empezar, el usuario cuenta con un perfil, en el que se encuentra un espacio llamado “blog” para que escriba todas las reflexiones que quiera compartir con sus amigos, y otro espacio llamado “Mi tablón”, dedicado a que estos últimos

puedan escribirle al usuario comentarios de forma pública que puedan ver el resto de usuarios –siempre y cuando el usuario lo permita-. Si se quiere escribir un mensaje de forma más privada, se puede hacer a través de la sección de “Mensajes” o del botón “Mensaje privado” en el perfil de cualquier otra persona.

Además de estos comentarios, un usuario puede subir a la red social tanto fotos como vídeos, y etiquetar a sus amigos en ellos, de tal forma que estas fotos y vídeos también puedan ser vistos a través de los perfiles de esos amigos.

Igualmente, un usuario puede unirse a diferentes eventos o páginas de información sobre algún tema, y recibir notificaciones sobre cualquier cambio o comentario que se escriba en ellos. En el caso de los eventos, el usuario también puede activar una opción llamada “Check-in” que permite que los usuarios vean a través de su estado –mensaje corto que un usuario puede publicar a modo de comentario personal- el lugar del evento.

Un usuario puede también compartir los sitios que suele frecuentar, y ver el resto de gente al que le gusta ese sitio y que también lo suele frecuentar. Otra información que el usuario puede compartir es su unión a una red social (universidades, empresas, etc.), y al igual que en el caso de los “Sitios”, el usuario también puede ver quienes forman parte de esa red, y de esta forma, encontrar viejas amistades.

En el perfil se encuentra además otros datos personales (tales como el sexo, la fecha de nacimiento, la ciudad y el estado civil –adecuado al target de usuarios de la web- del usuario) que puede ser consultado por cualquier persona que pueda acceder al perfil.

Por último, el usuario puede chatear con sus amigos que se encuentran conectados en ese momento a través de la herramienta “Chat”, y jugar a los diferentes “Juegos” presentes en la red social. Estas últimas aplicaciones, los juegos, contienen una alta carga social, ya que casi todos son juegos de al menos 2 jugadores en los que el usuario se puede enfrentar con un amigo o con cualquier otro usuario de la red social. Además, se muestran diferentes estadísticas en función de la puntuación de los usuarios que fomentan el uso de estas aplicaciones y de contactar con otros usuarios para utilizar dichos juegos.

Toda esta información puede ser limitada en función de la configuración de la privacidad que se le quiera dar a la cuenta. Un usuario puede bloquear ciertas acciones (ver el perfil, ver las fotos, descargarlas, enviar mensajes, etc.) en función del nivel de distancia a la que se encuentre una persona: puede escoger que la acción sea realizada por cualquier usuario de Tuenti, sólo por amigos de amigos, sólo los amigos, o que nadie la realice.

Además de esto, para que un usuario pueda publicar una información, según la política de privacidad de Tuenti, éste debe ser el propietario o titular de los derechos de los contenidos que vaya a subir a la web, no debe vulnerar ninguna ley de derecho a la intimidad y/o propiedad intelectual y se han debido obtener los permisos sobre datos de terceras personas.

Cabe destacar también que Tuenti no permite la publicación de direcciones URL o contenidos de la red social en lugares ajenos a éste. Esto implica que todo lo que es subido a Tuenti, únicamente podrá ser revisado desde dicha web y será ilegal visualizarlo u obtenerlo de otras páginas web. Si a esto le sumamos que las condiciones de uso prohíben “instar, pedir o solicitar a otros usuarios, a través del Servicio de TUENTI, datos personales o solicitar contraseñas o datos de carácter personal” (Tuenti, 2012), esto impide desarrollar aplicaciones basadas en la consulta de datos de un usuario.

En lo referido a la zona de desarrollo que se acaba de mencionar, al visitar dicho espacio, lo único que se encuentra en ella es cómo incluir un botón para compartir contenidos de una web en Tuenti: en ningún momento se informa acerca de la posibilidad de realizar aplicaciones para esta red social.

2.4.2. Twitter

Twitter es una red social basada en el microblogging, es decir, en mensajes cortos, tipo SMS de un móvil. Un usuario puede enviar un comentario a la red social – llamado *tweet* – con un máximo de 140 caracteres para indicar, por ejemplo, aquello que está haciendo, lo que le pasa, o para expresar sus sentimientos. Además, también se puede compartir enlaces, fotografías (siendo ambas direcciones abreviadas), o tu ubicación, siempre y cuando éstos, junto al mensaje que se quiera poner, no superen el límite máximo de caracteres.

Junto a esta información que puede compartir el usuario, éste también puede visualizar los *tweets* de otros usuarios siempre que esté siguiéndolos, es decir, siempre que se suscriba a sus comentarios. En ese momento, el usuario podrá ver los *tweets* de las personas a las que estás siguiendo de forma cronológica a cómo se hayan ido publicando. El usuario además puede contestar a los comentarios, creando así una conversación, o “*retwittear*” el comentario, es decir, compartir el *tweet* de otro usuario con la gente que sigue al usuario que lo “*retwittea*”.

Este es el funcionamiento básico de esta red social creada en julio de 2006 en San Francisco (Arrington, 2006). Twitter fue creada por la compañía Obvious Corp., que encontraba entre sus miembros a Evan Williams y Noah Glass, desarrolladores de otros sitios web como Odeo –directorio de RSS de audio y vídeo– (Wikipedia, 2011). Inicialmente sólo estaba en inglés, pero se fue traduciendo a diferentes idiomas, entre ellos el castellano, gracias a la ayuda voluntaria de diferentes usuarios de la red social (EFE, 2009).

Twitter es una red social que creció rápidamente, y que ha servido para estar también al corriente de las últimas noticias y temas del día gracias a los “Trending Topics”. Los “Trending Topics” –o TT–, son los términos de los que más se está hablando en ese momento en una región, en un país, o en el mundo entero, pudiendo ver los últimos comentarios de las personas que dijeron algo sobre el tema del momento. A esto le ayuda el “hashtag”, es decir, una cadena de texto o de varias

palabras concatenadas, precedidas por una almohadilla, que representa el tema que se trata en el *tweet*.

En esta red social se presenta menos información que en otras, ya que la información que el usuario proporciona es únicamente los comentarios que publica. La política de privacidad de Twitter (Twitter, 2011) menciona que “la mayor parte de la información que usted nos proporciona es información que usted nos está pidiendo que se haga pública”; con esto se puede determinar, que los contenidos de los comentarios pueden ser obtenidos y utilizados sin ningún tipo de problema, ya que además, a menos que así lo tenga configurado el usuario, toda esta información del perfil puede ser vista y accedida a través de cualquier motor de búsqueda.

Ya que se puede obtener una gran cantidad de información de los usuarios, esta red social podría ser de interés para realizar la aplicación. Al contrario que Tuenti, esta red social permite realizar diferentes aplicaciones en las que se integran las diferentes funcionalidades de Twitter (Twitter, 2012). Para ello utiliza diversas API's, entre las que se incluyen funciones de REST, de “streaming”, y de búsquedas en tiempo real.

2.4.3. Facebook

Facebook es la red social por antonomasia. Creada a principios de 2004 por el estudiante de la Universidad de Harvard, Mark Zuckerberg, esta red social se ha convertido en una de las webs más importantes en todo el mundo, debido a su gran volumen de visitas al día –según el sitio web alexa.com, que muestra información acerca de las visitas de los usuarios en los diferentes sitios de Internet, Facebook es la segunda web con mayor número de visitas al día en todo el mundo, sólo por detrás de Google (Alexa, 2012)- y su gran número de usuarios registrados, siendo líder en este aspecto entre todas las redes sociales del mundo.

El funcionamiento de esta red social es similar al que se ha descrito para la red social “Tuenti”, en el apartado “1.4.1 Tuenti”. Facebook permite buscar a cualquier persona y agregarla a la lista de amigos del usuario. La diferencia con respecto a ella es que mientras que Tuenti no deja ver todos los perfiles de los usuarios si estos lo tienen protegido, Facebook si permite ver todos los muros de los usuarios, o al menos, parte de ellos.

Según la política de privacidad de Facebook (Facebook, 2011), existe una cierta información que siempre va a ser pública: el nombre, las fotos de perfil, las redes a las que se está unido y el nombre e identificador de usuario. Una persona que tenga su perfil totalmente configurado para que ninguna información sea visible por quién no deba, tendrá expuesta al menos los datos que se acaban de mencionar.

Dentro del perfil de cada usuario, se encuentran diferentes apartados. Por un lado, está la pestaña “Muro”, donde se ven las últimas fotos del usuario, su información básica, sus últimas actividades realizadas en la red social (amigos nuevos, aplicaciones utilizadas, comentarios de otros amigos, etc.) y sus amigos. Por otro lado, se encuentra la pestaña “Información”, dónde se pueden encontrar todos

los datos pertenecientes al usuario, tales como las redes de formación y empleo (universidades, institutos, etc.), ideologías, deportes, ocio y otras actividades e intereses. Además, se puede obtener información básica, como la situación sentimental, e información de contacto (e-mail, teléfono, etc.). También hay un apartado dedicado a las “Fotos”; en él, se pueden ver todas las fotos que el usuario haya subido a Facebook, o que sus amigos hayan subido y le hayan etiquetado. Finalmente, hay otro apartado llamado “Amigos”, en el que se pueden encontrar todos los amigos del usuario.

Además de todo esto, un usuario puede unirse a diferentes grupos y páginas referentes a diversos temas e intereses comunes. Los usuarios también pueden utilizar un sinnúmero de aplicaciones y juegos para su entretenimiento. Al igual que se mencionó para la red social “Tuenti”, todas estas aplicaciones y juegos están cargados de un alto contenido social, teniendo que pedir ayuda a los amigos o competir contra ellos para avanzar en la aplicación.

En cuanto a la privacidad, todos los datos e información que se publiquen tendrán que ser configurados desde el apartado de privacidad, puesto que si éstos se configuran como datos públicos, será como si el usuario lo estuviera compartiendo con todo el mundo, y por tanto, podrían ser usados por cualquier persona. De hecho, cualquier dato público podrá ser consultado a través de la API de Facebook.

Esta red social es la que más facilidades da a la hora de crear aplicaciones. Tiene un apartado dentro de la web dedicada al desarrollo de aplicaciones (Facebook, 2012), en la que se puede obtener ayuda para insertar elementos de Facebook en una página web o aplicaciones de dispositivos móviles, o para diseñar y construir aplicaciones para Facebook. De hecho, tiene una API con la que se puede obtener cualquier dato necesario de una persona para utilizarla en la aplicación a crear.

En la ayuda para el desarrollo de aplicaciones también se incluyen pequeños tutoriales y consejos de cómo realizar una aplicación para Facebook. Como se puede observar analizando el contenido de la página de desarrollo, Facebook contiene multitud de herramientas para ayudar en la realización de una aplicación; de hecho, estas aplicaciones se pueden desarrollar en diferentes lenguajes de programación en función del entorno para el que se esté desarrollando (Android, iOS, PHP, etc.). Además, cuenta con sus propias API's de técnicas y lenguajes ya conocidos (REST, JavaScript, etc.), junto con otras desarrolladas por la propia red social –por ejemplo, Facebook cuenta con el lenguaje FQL, una especie de versión del lenguaje de programación de bases de datos SQL, orientado a la obtención de datos de usuarios de Facebook-.

2.4.4. Conclusiones

Tal y como se ha podido ver en estos apartados, todas las redes sociales contienen arquitecturas comunes en cuanto a su funcionamiento, y su éxito se basa en los contenidos extras que hacen que los usuarios no puedan despegarse de ellas – por ejemplo, la existencia de juegos y aplicaciones en Facebook con una alta carga social hacen que enganchen a la gente y les hagan participar más en esta red social que en otras-.

Sí que existe mayor diferencia entre las diversas políticas de privacidad y uso de los datos que hace cada red social, y sobre la posibilidad de crear aplicaciones sobre ellas. En estos aspectos, Tuenti es la más segura, puesto que sus datos no pueden ser sacados de la propia red social, y ni siquiera se pueden crear aplicaciones en ella. Twitter es un caso especial, puesto que en esta red social, los datos que se incluyen son menores, y no acarrear tanto peligro; no obstante, sí es cierto que si no se protegen, cualquier persona que esté navegando por Internet se puede encontrar con tus comentarios. Finalmente, Facebook es la que más problemas tiene en cuanto a privacidad se refiere, puesto que, por muy alto que sea su nivel de privacidad, siempre se podrá obtener algún dato del usuario; en cuanto a las aplicaciones se refiere, es la que más facilidades da a la hora de crearlas.

Una vez vistas las diferentes políticas de privacidad, la posibilidad de crear aplicaciones en las redes sociales, y las tecnologías que se pueden utilizar para ello, se puede concluir que la red social Facebook será la destinataria de crear la aplicación que en este proyecto se está exponiendo.

2.5. Aplicaciones en Facebook

En Facebook existen multitud de aplicaciones de diversa índole: juegos, noticias, calendarios, curiosidades, tests, etc. En total, más de 500.000 aplicaciones a fecha de enero de 2010 (Wikipedia, 2012). Esta gran cantidad de aplicaciones es debida a que la creación de éstas es accesible para todo el mundo a través de la zona para desarrolladores mencionada en el punto “1.4.3. Facebook”.

Como ya se dijo, el punto fuerte de estas aplicaciones es su carga social. La gran mayoría de las aplicaciones cuentan con una comparativa de los resultados de los compañeros en un test, o con la clasificación de los amigos en un juego según quien haya avanzado más en él. Además, otra posibilidad es que para avanzar en un juego, se tenga que pedir ayuda a los amigos.

Se pueden destacar algunas aplicaciones y juegos entre todas las existentes. Una de las que más éxito tiene en la actualidad es “The Sims Social”. Esta aplicación es una versión del mítico juego de ordenador “Los Sims”, con la variante de que se ha diseñado para una red social, en la cual se puede pedir ayuda a los amigos para seguir desarrollando la vida virtual del personaje.

El funcionamiento del juego es muy similar al juego de ordenador: se controla un personaje que tiene unas necesidades (diversión, hambre, etc.), y que realizando diversas acciones (ver la tele, cocinar, etc.) quedan satisfechas. La diferencia del juego de ordenador y éste, es que en el de la red social se puede visitar a los vecinos que en verdad son personas de la lista de amigos del usuario que también están jugando.

Además, en el juego existen misiones que se pueden ir completando realizando otras acciones en el juego, o conversando con vecinos del juego (es por ello que se hace más importante el tener amigos en la red social, aunque sea sólo por el juego). Para realizar estas acciones, hay veces en que los vecinos tienen que enviar objetos para poder cumplirlas, así como para tener más muebles en la casa. De esta forma, con este tipo de juegos, se consigue que un usuario se enganche a él, y por ende, a la red social.

En las imágenes 2 y 3 se pueden ver unas capturas de pantalla del juego, en la que los amigos solicitan ayuda y la visita del personaje a otros vecinos y su interacción con él.



Imagen 2. Amigos solicitando ayuda en el juego 'The Sims Social'.



Imagen 3. Personaje interactuando con un vecino.

Existen otras muchas aplicaciones de diferentes tipos: las hay para agrupar todas las fechas de cumpleaños de la lista de amigos en una sola aplicación, también hay aplicaciones que realizan preguntas sobre diversos aspectos de las personas que tengas agregadas, o hay aplicaciones decorativas que realizan, por ejemplo, un árbol de Navidad con las fotos de la lista de amigos. En la imagen 4 se puede observar un ejemplo de aplicación de preguntas sobre los amigos.

The screenshot shows a user interface for a social media application. At the top, a yellow banner reads "Você tem 5 novas respostas sobre você! Descubra o que seus amigos disse!". Below this is a progress bar labeled "5% completo". A green banner below the progress bar states: "Ganhar 1 Crédito por cada pergunta que responder. Ganhar 5 Bônus em Créditos por cada 25 pergunta que você responder." The main content area is divided into two parts. On the left, a card titled "75 - Pergunta sobre:" features a profile picture of a person named "Violeta". Below the name, there is a checkbox labeled "Publicar no mural de seu amigo(a)" which is checked, and a link "Pular este(a) amigo(a)". On the right, the question "Violeta provavelmente terá uma vida feliz?" is displayed. Below the question are three buttons: "Sim" (blue), "Não" (blue), and "Talvez" (grey).

Imagen 4. Aplicación que realiza preguntas sobre los amigos de un usuario.

En lo referente al tema que ocupa este documento, existen pocas aplicaciones que traten con grafos de búsqueda dentro de una red social. Todas las aplicaciones obtienen datos de cualquier usuario, puesto que son necesarios para realizar las diferentes acciones que componen la aplicación, pero pocas realizan labores de estadística o de grafos sobre estos datos. Una de las pocas aplicaciones que hay en Facebook referente a grafos sociales es “myFnetwork”, que elabora un grafo de toda la lista de amigos, y sus conexiones entre los que forman la lista de amigos si las hubiera, mostrando a éstos con un color más claro cuántas más conexiones tuvieran. En la imagen 5 se puede ver el resultado de uno de estos grafos.

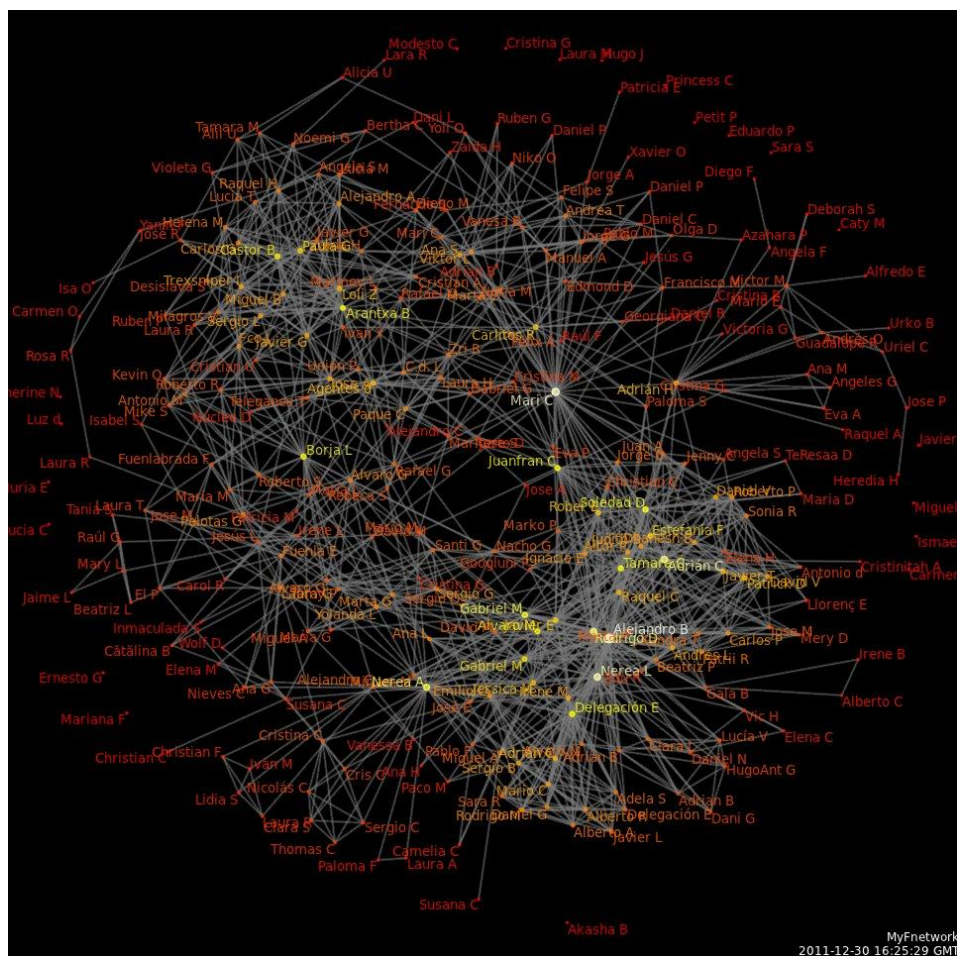


Imagen 5. Grafo de amistades.

2.5.1. Desarrollo de aplicaciones en Facebook

Facebook contiene una zona de desarrolladores (Facebook, 2012) con bastante información para los programadores. En esta sección, se puede encontrar tanto documentación para añadir plugins sociales a una web (añadir el típico botón “Me gusta”, añadir el registro a una web mediante Facebook, poder comentar en el perfil de un usuario desde la web, etc.), como datos para crear una aplicación en Facebook o utilizar la red social en un dispositivo móvil.

En cuanto a las aplicaciones para Facebook, la red social nos proporciona diferentes secciones en función de las múltiples opciones que se pueden configurar a la hora de crear una aplicación (notificaciones a otros usuarios, estadísticas, la forma de cargar la aplicación, etc.), además de diferentes consejos y tutoriales.

Para realizar la programación de las aplicaciones para Facebook, esta red social ofrece la posibilidad de utilizar alguna de sus 4 SDK disponibles. Dos de ellas están dedicadas a dispositivos móviles (Android e iOS), mientras que las otras dos están dedicadas a sitios web. Estas SDK se corresponden a los lenguajes JavaScript, que está más dedicado a las funcionalidades en el lado del cliente, y PHP, que está

más orientado a la conexión cliente-servidor. Debido a las características de la aplicación que se va a desarrollar, se utilizará la SDK de JavaScript para desarrollar las conexiones con el servidor de Facebook desde la aplicación.

Centrados en la SDK de JavaScript (Facebook, 2012), se pueden identificar diferentes métodos para obtener y manejar los diversos datos que se pueden obtener de Facebook. Existen 3 métodos principales:

- **FB.api:** Este método llama directamente a la API de Facebook para obtener los datos estructurados –que se detallarán más adelante- de una dirección concreta. Los parámetros que se pueden pasar a la función son la dirección a analizar, el método HTTP a utilizar (que por defecto es “GET”), los parámetros para la consultar, y la función para manejar la respuesta obtenida por el método. En la tabla 1 se muestra un ejemplo simple de este método:

```
FB.api('/me', function(response) {  
    alert(response.name);  
});
```

Tabla 1. Ejemplo de llamada al método FB.api

- **FB.init:** Este método permite inicializar el SDK de JavaScript con la aplicación que vamos a ejecutar. Entre los parámetros que se pueden pasar a este método destacan el identificador de la aplicación, para saber que aplicación debe inicializar, la URL del canal que apuntará a la dirección en la que se encuentre el SDK de JavaScript, y varios valores booleanos que permiten comprobar el estado del usuario, la posibilidad de utilizar cookies, y la posibilidad de analizar los datos XFBML –se explicará más adelante-.
- **FB.ui:** Este método permite generar diferentes elementos de comunicación entre la aplicación con los diferentes usuarios de la red social. Entre estos elementos, se pueden encontrar el aviso para añadir a un nuevo amigo, el mensaje de aceptación de permisos, o el envío de una notificación a un amigo. En este método entran como parámetros el objeto en el que se incluyen todos los atributos a incluir en el aviso, y la función que maneja dicho objeto. En la tabla 2 se muestra un ejemplo simple de este método:

```
FB.ui(  
  
  {  
  
    method: 'feed',  
  
    name: 'Facebook Dialogs',  
  
    link: 'http://developers.facebook.com/docs/reference/dialogs/',  
  
    picture: 'http://fbrell.com/f8.jpg',  
  
    caption: 'Reference Documentation',  
  
    description: 'Dialogs provide a simple, consistent interface for applications to interface with  
users.'  
  
  },  
  
  function(response) {  
  
    if (response && response.post_id) {  
  
      alert('Post was published.');  
    } else {  
  
      alert('Post was not published.');  
    }  
  
  }  
  
);
```

Tabla 2. Ejemplo de llamada al método FB.ui

Aparte de estos métodos, existen otros que complementan a los citados anteriormente. Por un lado, se encuentran los métodos de autorización: estos métodos permiten conocer el estado en el que se encuentra el usuario y los permisos que haya dado a cierta aplicación. Por otro lado, también se encuentran métodos para el manejo de eventos. Estos métodos se basan en la suscripción o baja de diferentes eventos, como puede ser el hacer clic en el botón “Me gusta” de una página de Facebook.

Otros métodos son utilizados para manejar el XFBML. XFBML es un lenguaje creado por Facebook para poder ejecutar diferentes funcionalidades dinámicamente en el lado cliente. Con este método, por ejemplo, podemos generar las imágenes de perfil de los usuarios en función al resultado de una función, y mostrarlo a continuación en la aplicación una vez haya sido inicializada ésta anteriormente. De igual forma, se puede hacer con los contadores de “Me gusta”, cajas de comentarios, etc.

Para finalizar, existen otra serie de métodos para manejar la pantalla en la que se representará la aplicación. Entre estos métodos, existen los que sirven para ajustar el tamaño de la pantalla, obtener información de la página, manejar el tiempo de carga, etc.

Junto a los SDK, existen otras herramientas que ayudan a la labor de los programadores, permitiendo probar códigos escritos en JavaScript y observar la estructura de datos que siguen las diferentes direcciones de Facebook. A continuación, se detallan su funcionamiento:

- **JavaScript Test Console:** Como ya se ha dicho, esta herramienta (Facebook, 2012) es una pequeña consola en la que se puede ejecutar diferentes códigos de JavaScript e insertando los diferentes métodos de Facebook que se han descrito anteriormente. Además de ejecutar un código cualquiera, esta consola te da la posibilidad de ejecutar diferentes ejemplos que te ayuden en la tarea de la codificación. Existen ejemplos para los diferentes métodos descritos anteriormente.

JavaScript Test Console

Tools > JavaScript Test Console

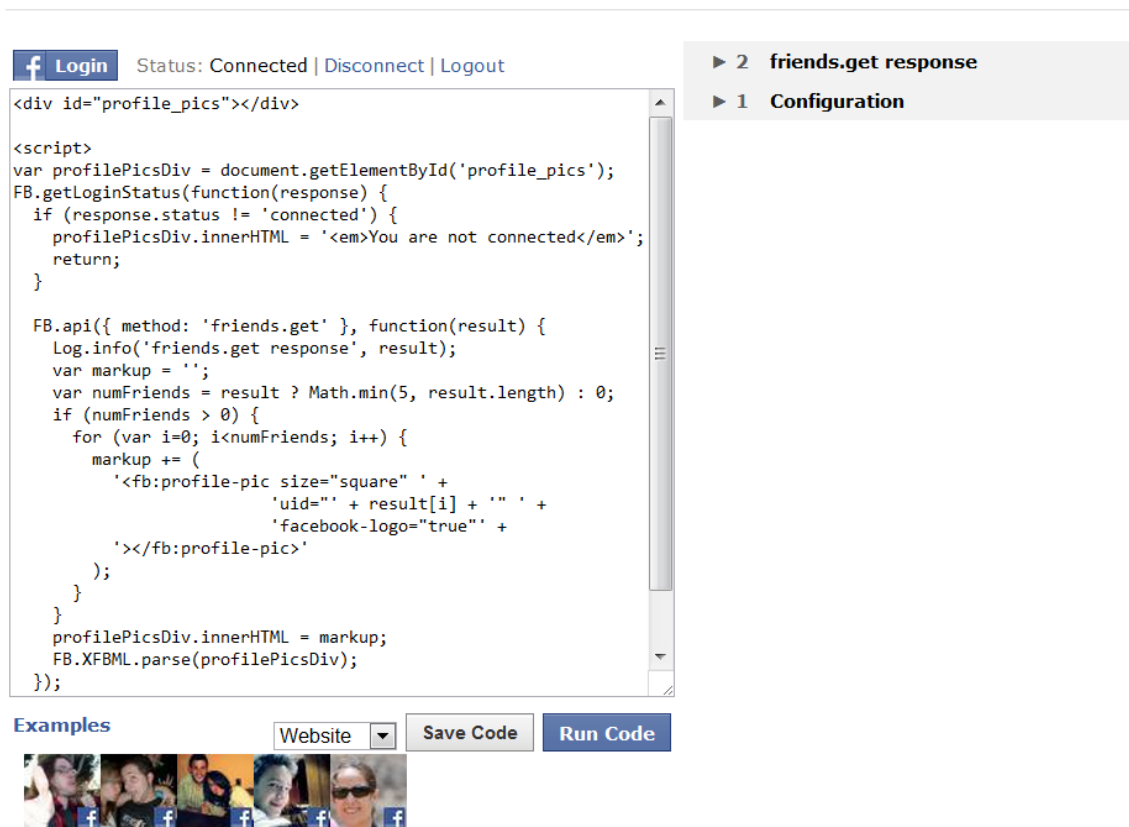


Imagen 6. JavaScript Test Console

En la imagen 6 se puede apreciar por un lado el editor de textos en el que poder escribir el código, la zona inferior a éste en el que se muestran los resultados, y a la derecha, un registro de resultados.

- Graph API Explorer: Esta herramienta (Facebook, 2012) permite analizar la estructura de datos de cualquier elemento que esté registrado en Facebook, desde un usuario hasta lugares, empresas y datos de la propia red social. Gracias a esta herramienta, se puede saber que formato tendrá el elemento que devuelva la función FB.api en base a una consulta realizada.

Graph API Explorer

Inicio > Herramientas > Graph API Explorer

Aplicación: [?] **Graph API Explorer**

Access Token: [Get Access Token](#)

GET **Enviar**

```

{
  "id": "1528837828",
  "name": "José Ángel Cuadrado Mingo",
  "first_name": "José Ángel",
  "last_name": "Cuadrado Mingo",
  "link": "http://www.facebook.com/joseangel.cuadradomingo",
  "username": "joseangel.cuadradomingo",
  "hometown": {
    "id": "108341312519779",
    "name": "Leganés"
  },
  "location": {
    "id": "108341312519779",
    "name": "Leganés"
  },
  "work": [
    {
      "employer": {
        "id": "171416199583662",
        "name": "Laboratorio del Departamento de Informática - UC3M"
      },
      "location": {
        "id": "104057716296112",
        "name": "Leganés"
      },
      "position": {
        "id": "143854602308589",
        "name": "Becario"
      },
      "start_date": "2010-02",
      "end_date": "2012-02"
    },
    {
      "employer": {
        "id": "107665539263809",
        "name": "CDE Tecnocasa Leganenses"
      },
      "location": {
        "id": "104057716296112",
        "name": "Leganés"
      },
      "position": {
        "id": "143854602308589",
        "name": "Becario"
      },
      "start_date": "2010-02",
      "end_date": "2012-02"
    }
  ]
}

```

Conexiones

accounts activities adaccounts albums
 apprequests books checkins events family
 feed friendlists friendrequests friends games
 groups home inbox interests likes links
 movies music mutualfriends notes
 notifications outbox payments permissions
 photos picture posts scores statuses
 tagged television updates videos

Fields

id
The user's Facebook ID. No 'access_token' required.
'string'.

name
The user's full name. No 'access_token' required.
'string'.

first_name
The user's first name. No 'access_token' required.
'string'.

middle_name
The user's middle name. No 'access_token' required.
'string'.

last_name
The user's last name. No 'access_token' required.
'string'.

gender
The user's gender: 'female' or 'male'. No 'access_token' required. 'string'.

locale
The user's locale. No 'access_token' required. 'string' containing the ISO Language Code and ISO Country Code.

languages
The user's languages. 'user_likes' - 'array' of objects containing language 'id' and 'name'.

link
The URL of the profile for the user on Facebook. No

Imagen 7. Graph API Explorer

En la imagen 7, se pueden identificar las diferentes secciones del explorador. En la parte superior, se encuentra la dirección de la que se quiere hacer la consulta (a través de su identificador o nombre de usuario), y el “access token”, que es una cadena de texto que representa los permisos que se han concedido a una aplicación. En la parte central se encuentra la respuesta a la petición enviada; en ella, se puede ver los diferentes campos de una persona, divididas en una serie de objetos y arrays. A la derecha de este cuerpo, se encuentran las conexiones que dependen de la dirección URL en que se encuentra, y la explicación de los diferentes campos con valores que se pueden presentar en la zona central.

2.5.2. Ejemplo de creación de una aplicación en Facebook

Crear una aplicación sencilla en Facebook, como puede ser mostrar la foto del perfil de algún usuario, no es demasiado complicado. Se va a proponer un ejemplo de creación de una aplicación sencilla en Facebook para que muestre la foto del perfil de los amigos de un usuario.

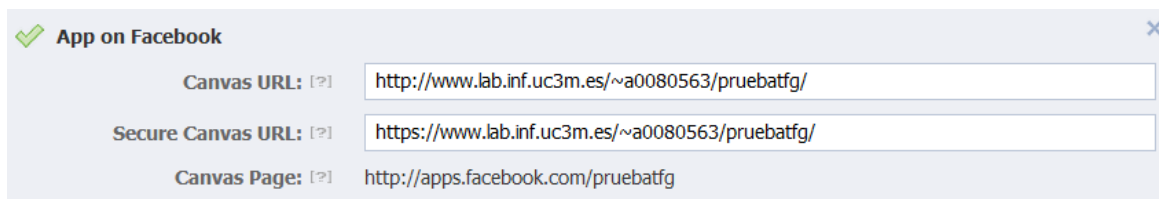
Utilizando la SDK de JavaScript, para crear esta aplicación basta con utilizar código HTML y añadirle esta funcionalidad. Pero antes de crear el código, se debe dar de alta nuestra aplicación en Facebook. Para ello, hay que acceder a la zona de desarrolladores de Facebook, y entrar en el apartado de aplicaciones (Facebook, 2012). En este apartado, se pulsa en el botón “Create New App”, y saldrá algo similar a la imagen 8.

Imagen 8. Creación de una nueva aplicación

En esta pantalla, se puede rellenar el nombre de la aplicación, su “namespace” –esto es, el nombre de la aplicación que saldrá en la URL-, y si se requiere un lugar para albergar la aplicación. Una vez hecho esto, al hacer clic en “Continuar”, mostrará una pantalla como la que se muestra en la imagen 9.

Imagen 9. Configuración básica de la aplicación

En esta pantalla cabe destacar la información que hay junto al nombre: la “App ID” y la “App Secret”. Estos datos son necesarios cuando se vaya a inicializar la aplicación. En la parte inferior, se encuentran diferentes formas de integrar nuestra aplicación en Facebook. Para el caso que ocupa, utilizaremos la de “App on Facebook”, tal y como se muestra en la imagen 10.



The image shows a configuration window titled "App on Facebook" with a green checkmark icon. It contains three input fields, each with a help icon (?) to its left:

- Canvas URL:** `http://www.lab.inf.uc3m.es/~a0080563/pruebatfg/`
- Secure Canvas URL:** `https://www.lab.inf.uc3m.es/~a0080563/pruebatfg/`
- Canvas Page:** `http://apps.facebook.com/pruebatfg`

Imagen 10. Integración de la aplicación en Facebook

En este apartado, simplemente tenemos que añadir la dirección URL en la que se encontrará alojada la aplicación. La conexión segura es opcional, pero si no se indica, los usuarios de la red social que estén navegando de forma segura, no podrán entrar en la aplicación.

Con estas acciones bastarían para poder tener disponible la aplicación. No obstante, en esta zona, también se podrían hacer configuraciones extras para seleccionar los permisos que un usuario debe aceptar para utilizar la aplicación, para seleccionar las restricciones de edad del usuario o para configurar el tamaño de la pantalla en la que se muestre, entre otras opciones.

A continuación, se creará el código para la aplicación. Para ello, se utilizará la SDK de JavaScript. Para poder cargarla correctamente e inicializar la aplicación, se debe añadir el siguiente trozo de código proporcionado por Facebook al inicio de la etiqueta “<body>”, tal y como se muestra en la web del SDK de JavaScript (Facebook, 2012), en la sección de “Loading”. Para que funcione el canal de inicialización de Facebook, hay que añadir además una referencia a un script de Facebook, descrito en la sección “Channel File” de dicha web.

Posteriormente, se crea una capa en la que se mostrarán las fotografías. En él, se añade una función de JavaScript en el que se identifica a la capa, y se llama a una función de Facebook para reconocer el estado en el que se encuentra el usuario; si está conectado, llama a la API de Facebook para obtener la lista de amigos, y a continuación, se analizan los contenidos para mostrarlos en la web las imágenes de perfil a través del XFBML.

Finalmente, se llama a este método de JavaScript después de inicializar la aplicación (justo tras el comentario que te indica de añadir el código) para que ejecute esta función, y estará terminada la aplicación.

El código final resultante será el siguiente:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

  <head>

    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

    <title>Prueba TFG | Facebook</title>

    <script src="//connect.facebook.net/en_US/all.js"></script>

  </head>

  <body>

    <div id="fb-root"></div>

    <script>

      window.fbAsyncInit = function() {

        FB.init({

          appId      : '424793490875025', // App ID

          channelUrl  : 'http://www.lab.inf.uc3m.es/~a0080563/pruebatfg/channel.html', // Channel
File
          status      : true, // check login status

          cookie      : true, // enable cookies to allow the server to access the session

          xfbml       : true // parse XFBML

        });

        // Additional initialization code here

        imagenes();

      };

      // Load the SDK Asynchronously

      (function(d) {

        var js, id = 'facebook-jssdk', ref = d.getElementsByTagName('script')[0];

        if (d.getElementById(id)) {return;}

        js = d.createElement('script'); js.id = id; js.async = true;

        js.src = "//connect.facebook.net/en_US/all.js";

        ref.parentNode.insertBefore(js, ref);

      }(document));

    </script>

    <div id="profile_pics"></div>

    <script>

      function imagenes(){

        var profilePicsDiv = document.getElementById('profile_pics');
```

```
FB.getLoginStatus(function(response) {  
  
    if (response.status != 'connected') {  
  
        profilePicsDiv.innerHTML = '<em>You are not connected</em>';  
  
        return;  
  
    }  
  
    FB.api({ method: 'friends.get' }, function(result) {  
  
        var markup = '';  
  
        if (result.length > 0) {  
  
            for (var i=0; i<result.length; i++) {  
  
                markup += ('<fb:profile-pic size="square" uid="' + result[i] + '"  
facebook-logo="true"></fb:profile-pic>');  
  
            }  
  
        }  
  
        profilePicsDiv.innerHTML = markup;  
  
        FB.XFBML.parse(profilePicsDiv);  
  
    });  
  
});  
  
}  
  
</script>  
  
</body>  
  
</html>
```

Tabla 3. Código de ejemplo finalizado

El resultado final de esta aplicación sería algo similar a la imagen 11.

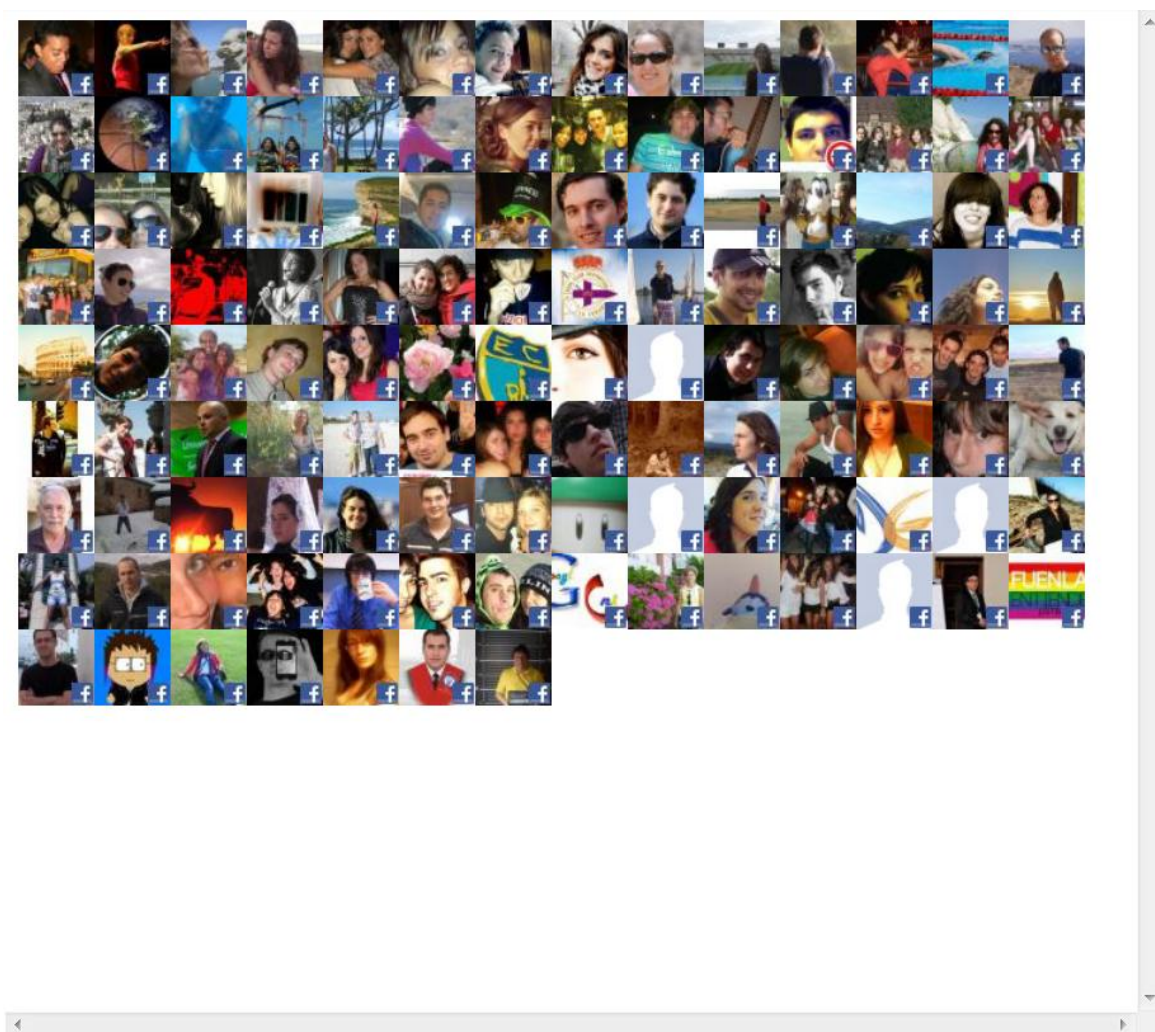


Imagen 11. Ejemplo del funcionamiento de la aplicación de ejemplo

2.6. Librerías externas

Aparte de las llamadas a la API de Facebook, serán necesarias otras librerías externas para poder realizar todas las funcionalidades requeridas, como por ejemplo, dibujar los grafos en la aplicación.

Al estar realizando una aplicación web, en la que están involucradas las tecnologías HTML y JavaScript, lo idóneo sería encontrar una librería que permitiera dibujar y manejar los grafos con lenguaje JavaScript para que interpretara correctamente las salidas de los métodos de Facebook.

Dentro de estas características se encuentra la librería Protovis (Stanford Visualization Group, 2011), una librería en JavaScript con multitud de ejemplos de diferentes gráficos y una API bien documentada creada por el “Stanford Visualization Group” de la Universidad de Stanford.

Protovis permite, a través de una serie de objetos de datos, construir cualquier gráfico con nodos o barras, de una forma rápida y ligera. Además, esta librería es open-source, por lo que no requiere gasto alguno para el proyecto.

Como ejemplo, se puede ver en la imagen 12 un grafo con diferentes nodos que se pueden manejar haciéndolo más grande o más pequeño, moviendo los nodos, arrastrando el grafo y viendo el nombre correspondiente de cada nodo posando el cursor sobre los diferentes nodos. Este ejemplo viene descrito en la zona de ejemplos de la web (Stanford Visualization Group, 2011).

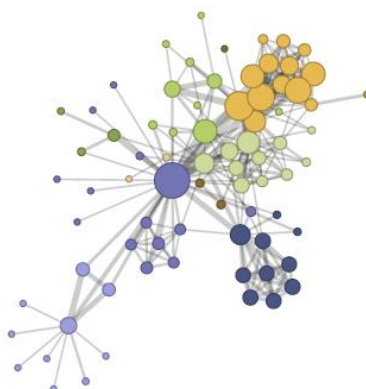


Imagen 12. Grafo generado con Protovis

Como único punto negativo de esta librería es que actualmente está parado su desarrollo ya que el grupo creador está trabajando en otra nueva librería mejorada de grafos llamada D3 (Stanford Visualization Group, 2012).

2.7. Conclusiones

En este proyecto, se realizará una búsqueda de relaciones extraídas de una red social. Se ha estudiado lo que es una red social y las diferentes redes sociales que existen, centrando la atención en las más importantes en el territorio español: Tuenti, Twitter y Facebook.

Mirando las diferentes características de estas tres redes sociales, finalmente se eligió a Facebook para realizar la aplicación de búsqueda de relaciones, ya que tenía un gran número de usuarios y había bastantes facilidades para realizar aplicaciones sobre este entorno.

Además, se utilizará una librería externa llamada Protovis para generar los grafos con los que mostrar los resultados obtenidos por el buscador.

A continuación, se detallarán los diferentes aspectos necesarios para desarrollar la aplicación que ocupa en este documento.

3. Estudio de Viabilidad del Sistema

En este apartado, se tratará el estudio sobre la posibilidad de crear la aplicación que ocupa en este documento.

3.1. Introducción

En el Estudio de Viabilidad del Sistema se analizarán las diferentes necesidades para crear el proyecto y poder realizarlo en un corto período de plazo en función de las diferentes restricciones del entorno (técnicas, legales, económicas, etc.).

El sentido de realizar este estudio viene dado por recoger las necesidades del cliente para después poder analizar los diferentes requisitos y casos de uso de la aplicación que propone, así como su alcance, teniendo en cuenta las posibles restricciones existentes.

Una vez realizado este estudio, se planteará una solución final en la que se resumirá la información necesaria para que, a partir de ésta, se realice el análisis, diseño e implementación de la aplicación.

3.2. Alcance del sistema

En este punto, se realizará una descripción general del problema planteado por el cliente, anotando todos los datos que pudieran ser de utilidad para convertirlos finalmente al proceso de desarrollo, conociendo de esta forma los requisitos y restricciones que debe albergar la aplicación que se desarrollará.

Analizando la información que proporciona el cliente, éste nos solicita una aplicación para una red social –sin especificar cuál- en la que se extraigan las relaciones existentes entre diferentes usuarios en función a una búsqueda, y mostrando dichas relaciones a través de un grafo. Entrando en más profundidad en la información proporcionada, se pueden encontrar los siguientes datos:

- Un usuario de la red social en la que se desarrolle la aplicación podrá buscar una cierta información a través de la aplicación. Esta búsqueda será realizada analizando los contenidos de los usuarios más próximos a él (amigos, seguidores, etc.) en función del grado de profundidad de ésta. Esta profundidad vendrá determinada en función de si el usuario quiere consultar los datos de sus amigos o también quiere consultar los datos de los amigos de sus amigos.
- El sistema obtendrá una serie de resultados en función a los parámetros de búsqueda que se hayan seleccionado, mostrando dichos resultados a través de un grafo, y mostrando los mejores de estos resultados en una lista.
- El grafo en el que se representan los resultados obtenidos por el buscador debe ser manejable, pudiendo, por ejemplo, aumentar o disminuir el tamaño de éste, o mostrar el nombre de la persona que represente cada nodo del grafo.
- La aplicación deberá contar además con una sección de ayuda para garantizar al usuario el correcto funcionamiento de la aplicación, y deberá poder ser utilizada también desde una conexión segura, para proteger los datos del propio usuario.
- Los datos que se obtengan de las personas próximas al usuario de la aplicación sólo se podrán obtener de la red social en la que se ejecute la aplicación, y en función del nivel de privacidad que tengan esas personas en dicha red social.
- La aplicación deberá poder ejecutarse y visualizarse correctamente a través de los navegadores Internet Explorer, a partir de su versión 9, Mozilla Firefox, a partir de su versión 4, y Google Chrome, a partir de su versión 19.

3.2.1. Identificación de los interesados en el sistema (stakeholders)

En este punto, se identifican las diferentes personas y entidades que pueden estar interesados de alguna forma en el proyecto. Se pueden identificar los siguientes interesados:

- Cliente: Es la persona que ha solicitado realizar este proyecto. En este caso, el cliente es José Arturo Mora Soto, profesor perteneciente al departamento SEL (Software Engineering Lab) de la Universidad Carlos III de Madrid
- Usuarios finales: Todas aquellas personas que sean usuarios de la red social Facebook y que puedan utilizar la aplicación.
- Tutor del proyecto: El tutor del proyecto será el encargado de dirigir, organizar y ayudar al autor de proyecto, así como supervisar las tareas que haga. Este papel recaerá sobre el cliente José Arturo Mora Soto.
- Autor del proyecto: José Ángel Cuadrado Mingo será el autor del proyecto, siendo además el máximo responsable de su realización y correcta finalización.

3.2.2. Planificación

En este apartado se analiza la duración total del proyecto, usando una planificación para realizar cada tarea en unas fechas previstas. Este proyecto se corresponde con la asignatura “Trabajo Fin de Grado”, con un valor de 12 créditos ECTS. Estos créditos equivalen a un total de 300 horas trabajadas por parte del alumno para poder realizarlo.

Teniendo en cuenta el total de horas, éstas se distribuirán a lo largo del cuatrimestre hasta el día máximo de entrega de la documentación, y distribuyendo también las horas en función del esfuerzo que reúne cada tarea del documento que aquí se detalla.

El resultado de esta planificación es realizar un proyecto en 300 horas, distribuidos entre los días 23 de Enero de 2012 y el 15 de Junio de 2012, es decir, en un total de 145 días, sin contar fines de semana.

En la tabla 4 se puede observar en detalle las diferentes tareas que se realizarán a lo largo del proyecto en las fechas indicadas, mientras que en la imagen 13, se observa la ocupación de las tareas en los diferentes días. De igual forma, se podrá ver en el “Anexo A” el diagrama de Gantt completo con las tareas que contiene los datos de la planificación.

Tarea	Subtarea	Duración (días)	Fecha inicio	Fecha fin
Introducción		5	23/01/12	27/01/12
Estado del arte	Estudio sobre redes sociales	5	23/01/12	27/01/12
	Estudio sobre privacidad	5	30/01/12	03/02/12
	Tecnologías de Facebook	5	06/02/12	10/02/12
Estudio de Viabilidad del Sistema	Alcance y situación actual	5	06/02/12	10/02/12
	Requisitos de usuario	5	13/02/12	17/02/12
Análisis del sistema	Definición del sistema	4	20/02/12	23/02/12
	Requisitos software	10	24/02/12	08/03/12
	Casos de uso	8	09/03/12	20/03/12
	Estructura del sistema	4	21/03/12	26/03/12
	Interfaces de usuario	4	27/03/12	30/03/12
Diseño del sistema	Arquitectura del sistema	5	19/03/12	23/03/12
	Arquitectura de soporte	5	26/03/12	30/03/12
	Diseño físico de datos	8	02/04/12	11/04/12
	Requisitos de implantación	2	12/04/12	13/04/12
Implementación del sistema	Interfaz de la aplicación	7	02/04/12	10/04/12
	Algoritmo de búsqueda	29	11/04/12	21/05/12
	Muestra de resultados	14	22/05/12	08/06/12
Verificación y validación del sistema	Plan de pruebas	3	28/05/12	30/05/12
	Informe de pruebas	5	31/05/12	06/06/12
	Pruebas de usuario	2	07/06/12	08/06/12
Conclusiones y líneas futuras		5	11/06/12	15/06/12

Tabla 4. Planificación

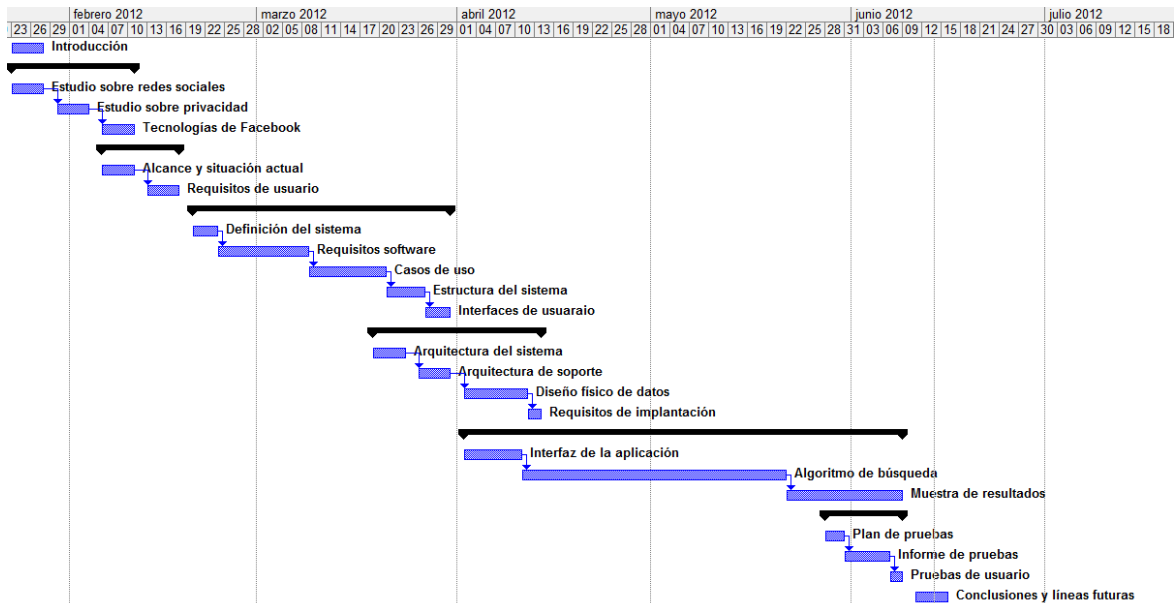


Imagen 13. Diagrama de Gantt de la planificación

Tal y como se puede ver en la tabla 4, se han identificado los puntos existentes en este documento como las fases a realizar en el proyecto, tal y como se describió en el punto “1.3. Estructura del documento”. Dentro de cada una de estas fases, también se encuentran los principales puntos a desarrollar en cada una de ellas.

3.2.3. Presupuesto

En este apartado, se calcula el presupuesto necesario para llevar a cabo este proyecto. Para ello, se tendrán en cuenta diferentes aspectos.

Por un lado, se tendrá que tener en cuenta el gasto de personal, que estará representado por el coste del tutor y del autor. Para calcular estos gastos, se utilizarán los salarios anuales de un ingeniero software senior y de un ingeniero software junior en Facebook, obtenidos a través de una reciente noticia (20minutos.es, 2012).

Según esta noticia, un ingeniero software senior cobra un total de 107.533 € anuales; para el proyecto que aquí se desarrolla, se estima que se realizará una reunión semanal, de una hora de duración, desde el comienzo del proyecto, por lo que trabajará un total de 21 horas, que equivale a un total de 1.433,77 €. En cuanto al ingeniero software junior, éste cobra un total de 81.155 € anuales; según lo establecido en el apartado “3.2.2. Planificación”, este puesto trabajará un total de 300 horas, que son equivalentes a un total de 15.487,09 €.

Además de esto, se tendrá en cuenta el coste derivado del uso de los diferentes equipos y productos necesarios para desarrollar el proyecto. En este apartado, tendremos en cuenta el coste amortizado del ordenador de desarrollo y del paquete ofimático –Microsoft Office Professional- para desarrollar toda la documentación. En total, se contará un total de 155 € en este aspecto.

Finalmente, sumando a esto un 20% de tasa de costes indirectos, el total asciende a 20.491,03 €. Este total estará desglosado en el “Anexo B”.

3.3. Situación actual

En el apartado “2. *Estado del Arte*” se hizo un análisis entre las diferentes redes sociales con más éxito en España, y seleccionando a Facebook como la red social más adecuada para hacer aplicaciones, se analizó un poco algunas aplicaciones existentes en esta red social.

En ese análisis, se comprobó que la gran mayoría de las aplicaciones existentes eran juegos sociales, en los que los usuarios debían colaborar y/o competir con otros para poder avanzar en el juego.

Analizando con detenimiento las aplicaciones, se pueden separar en dos grupos: los juegos sociales masivos, como por ejemplo “The Sims Social” o “Farmville”, y los juegos simples y pequeñas aplicaciones. La aplicación que se quiere desarrollar se encontraría en este segundo grupo.

Para realizar una aplicación de este último tipo, no se necesita una complejidad de desarrollo demasiado elevada, y tampoco se necesita un servidor demasiado potente, puesto que la aplicación no requiere que los usuarios la utilicen todos los días, si no sólo cuando necesiten realizar una consulta. Además, tampoco tiene un entorno gráfico que requiera una gran potencia para su correcto funcionamiento.

Debido a esto, se buscará una tecnología adecuada para el desarrollo de la aplicación. Como ya hicimos en el punto “2. *Estado del Arte*”, se analizaron las diferentes tecnologías que permite Facebook para desarrollar aplicaciones utilizando su base de datos. Finalmente, escogimos utilizar JavaScript ya que al no hacer falta ninguna conexión con el servidor de la aplicación, es más apropiada para manejar datos que se utilicen únicamente en el entorno del cliente.

Junto a esta tecnología, se utilizarán otras librerías que permitan construir grafos a partir de los resultados que se hayan obtenido al buscar entre los datos de las personas cercanas a un usuario. Para realizar esto, se usará una librería de JavaScript llamada Protovis que permite diferentes opciones a la hora de realizar y manejar grafos.

Como se ha comentado anteriormente, no se necesita un servidor demasiado potente para soportar la aplicación que se quiere desarrollar, por lo que se opta por almacenarla en un servidor dependiente del Laboratorio del Departamento de Informática de la Universidad Carlos III de Madrid. Este servidor contará con un sistema operativo Linux Debian Squeeze, procesador Intel Core i7 a 2.93 GHz y 3.5 GB de memoria RAM.

Finalmente, una vez analizada la situación actual de las aplicaciones en Facebook tanto en este apartado como en el punto “2. *Estado del Arte*”, se puede concluir que es una idea original y útil, ya que no existe una aplicación que te permita buscar cualquier contenido en las relaciones con los amigos y amigos de amigos de un usuario y mostrar los resultados de forma visual a través de un grafo.

3.4. Definición de requisitos de usuario

En este apartado, se extraerán los diferentes requisitos obtenidos por parte del cliente al describir el problema en cuestión para realizar el proyecto que se presenta en este documento. De esta forma, se podrá conocer realmente las funcionalidades que el cliente quiere obtener del proyecto.

A continuación, se realizará una lista detallada de los requisitos de usuario que se han extraído. La definición de estos requisitos muestra lo que el cliente quiere y/o necesita conseguir. Estos requisitos se pueden dividir en dos tipos:

- Requisitos de capacidad: Son aquellos requisitos que representan lo que un usuario necesita para conseguir un objetivo marcado.
- Requisitos de restricción: Son aquellos requisitos que representan las restricciones que tiene un usuario a la hora de lograr el objetivo marcado.

Para que los requisitos sean recogidos de la manera más clara posible y mantener una estructura de los mismos, se utilizará la plantilla mostrada en la tabla 5.

Código Requisito			
Nombre:			
Prioridad:	Alta/Media/Baja	Estabilidad:	Alta/Media/Baja
Necesidad:	Esencial/Deseable/Opcional		
Claridad:	Alta/Media/Baja	Verificabilidad:	Alta/Media/Baja
Descripción:			

Tabla 5. Plantilla de los requisitos de usuario

A continuación, se especificará el contenido y significado de los campos pertenecientes a la tabla 5.

- Código Requisito: Cada requisito incluirá una identificación para facilitar su traza para tareas subsiguientes. Dicho código seguirá el estándar RU-C-XX para los requisitos de capacidad y RU-R-XX para los requisitos de restricción, donde XX corresponde a un número ordinal.
- Nombre: Expresa en pocas palabras el significado del requisito.
- Prioridad: Indica la prioridad de desarrollo de un determinado requisito. Este campo puede tomar los valores “Alta”, “Media” y “Baja”.
- Estabilidad: Indica la posibilidad de que el requisito cambie a lo largo del desarrollo de la aplicación. Cuanto más alto, menor posibilidad de cambio.

- **Necesidad:** Indica el nivel de necesidad del requisito en el sistema. Este campo puede tomar los valores “Esencial”, “Deseable” u “Opcional”.
- **Claridad:** Indica la facilidad de comprensión del significado de un determinado requisito. Este campo puede tomar los valores “Alta”, “Media” o “Baja”.
- **Verificabilidad:** Indica la capacidad de verificación por parte del cliente de que se cumple un determinado requisito en el sistema final. Este campo puede tomar los valores “Alta”, “Media” o “Baja”.
- **Descripción:** Breve especificación de un determinado requisito mediante el uso de texto explicativo.

3.4.1. Requisitos de capacidad

RU-C-01			
Nombre:	La aplicación se ejecutará sobre una red social.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	La aplicación que se va a desarrollar debe poder ejecutarse sobre una red social, para que de esta forma haya más facilidad de obtener datos y relaciones con otros usuarios, así como tener mayores probabilidades de extender el uso de la herramienta. La red social escogida es escogida por el autor del proyecto, siendo indiferente para el cliente.		

Tabla 6. RU-C-01

RU-C-02			
Nombre:	El usuario podrá buscar información sobre una determinada característica.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Media
Descripción:	La aplicación consiste en un buscador que analiza los datos de las diferentes personas cercanas al usuario que busca la información. Por ello, el usuario podrá buscar cualquier término o conjunto de términos que quiera para encontrar las personas cercanas resultantes que devuelva la aplicación.		

Tabla 7. RU-C-02

RU-C-03			
Nombre:	El usuario podrá buscar la información entre sus amigos y los amigos de sus amigos.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Media
Descripción:	La aplicación podrá analizar los datos buscados por el usuario entre las personas más cercanas al usuario (amigos o seguidores), o también incluir a las personas más cercanas a estos amigos/seguidores del usuario.		

Tabla 8. RU-C-03

RU-C-04			
Nombre:	El sistema mostrará un aviso de que se está realizando la búsqueda.		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Cuando el usuario haya pulsado en el botón “Buscar”, el sistema mostrará algún tipo de aviso que sirva de referencia al usuario para saber que la aplicación ha empezado a buscar.		

Tabla 9. RU-C-04

RU-C-05			
Nombre:	El sistema sacará como resultado las personas que cumplan el criterio de búsqueda introducido por el usuario.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Una vez buscados los datos introducidos por el usuario, las personas que hayan obtenido similitudes con esta búsqueda serán almacenadas para mostrárselas posteriormente al usuario.		

Tabla 10. RU-C-05

RU-C-06			
Nombre:	El sistema mostrará los resultados a través de un grafo.		
Prioridad:	Alta	Estabilidad:	Media
Necesidad:	Esencial		
Claridad:	Media	Verificabilidad:	Alta
Descripción:	Los resultados extraídos por la aplicación serán mostrados en un grafo, siendo el nodo principal el propio usuario, y el resto de nodos que salgan desde el principal serán los amigos/seguidores del usuario. En caso de que también se quieran obtener resultados de los amigos de amigos, estos nodos serán unidos al amigo que tienen en común el usuario y la persona de segundo nivel.		

Tabla 11. RU-C-06

RU-C-07			
Nombre:	El sistema mostrará una lista con los mejores resultados de la búsqueda.		
Prioridad:	Media	Estabilidad:	Media
Necesidad:	Opcional		
Claridad:	Media	Verificabilidad:	Alta
Descripción:	Los mejores resultados extraídos por la aplicación serán mostrados en una lista junto al grafo, en el que se podrá identificar a cada persona coincidente con la búsqueda del usuario.		

Tabla 12. RU-C-07

RU-C-08			
Nombre:	El usuario podrá manejar el grafo.		
Prioridad:	Media	Estabilidad:	Media
Necesidad:	Deseable		
Claridad:	Media	Verificabilidad:	Alta
Descripción:	El grafo generado con los resultados de la búsqueda podrá ser manipulado por el usuario para mejorar su experiencia con la aplicación, de tal forma que podrá mover el grafo y los nodos por la zona delimitada para ello, así como obtener el nombre de cada propietario del nodo y alejar o acercar el grafo.		

Tabla 13. RU-C-08

RU-C-09			
Nombre:	El usuario podrá consultar una sección de ayuda.		
Prioridad:	Baja	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	El usuario tendrá una sección de ayuda en la que podrá consultar sus dudas para aclararlas fácilmente.		

Tabla 14. RU-C-09

RU-C-10			
Nombre:	El usuario podrá ejecutar la aplicación sobre una conexión segura.		
Prioridad:	Media	Estabilidad:	Media
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Para que todos los usuarios puedan manejar la aplicación de forma segura, se realizará una vía para que dicha herramienta pueda ser usada, por los usuarios que lo deseen, desde una conexión segura.		

Tabla 15. RU-C-10

3.4.2. Requisitos de restricción

RU-R-01			
Nombre:	La información buscada ha de obtenerse únicamente de la red social en la que se desarrolle la aplicación.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Baja
Descripción:	Al tratarse de una aplicación de búsqueda de información, se debe asegurar que la información sólo procede de la red social escogida para albergar nuestra aplicación, y así no tener problemas legales por violaciones de la privacidad.		

Tabla 16. RU-R-01

RU-R-02			
Nombre:	La información que se puede obtener dependerá del nivel de privacidad que tenga establecido cada persona.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Baja
Descripción:	Al tratarse de una aplicación de búsqueda de información, se debe asegurar que la información de los usuarios sólo se puede obtener si la privacidad del usuario lo permite, y así no tener problemas legales por violaciones de la privacidad.		

Tabla 17. RU-R-02

RU-R-03			
Nombre:	La aplicación ha de verse correctamente en Internet Explorer 9 –y superiores-, Mozilla Firefox 4 –y superiores-, y Google Chrome 19 –y superiores-.		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Estos son los navegadores que más usuarios utilizan, por lo que su correcta visión en ellos ha de ser esencial.		

Tabla 18. RU-R-03

RU-R-04			
Nombre:	La aplicación ha de verse correctamente en una resolución 1280x1024.		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Esta resolución se adecúa al tamaño de largo de la aplicación para su correcta visualización de una sola vez.		

Tabla 19. RU-R-04

RU-R-05			
Nombre:	La aplicación debe estar escrita en español.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Para mejorar la comprensión de la aplicación en el país en el que se ha desarrollado, esta herramienta tendrá que utilizar el español tanto en la interfaz como en la codificación.		

Tabla 20. RU-R-05

RU-R-06			
Nombre:	El sistema tratará de tener un buen rendimiento		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Opcional		
Claridad:	Alta	Verificabilidad:	Baja
Descripción:	Para ayudar al usuario en su experiencia, se intentará que la aplicación tenga un buen rendimiento y no tarde demasiado tiempo en realizar las consultas.		

Tabla 21. RU-R-06

3.5. Estudio de alternativas de solución

Después de analizar los diferentes aspectos técnicos que puede tener la aplicación tanto en el punto “2. *Estado del Arte*” como en este apartado, decidimos que la aplicación será realizada para la red social Facebook. La aplicación estará realizada en una web HTML alojada en el servidor del Laboratorio del Departamento de Informática de la Universidad Carlos III de Madrid, siendo este un servidor gratuito y fiable.

La forma de obtener datos de los usuarios de Facebook se podría hacer de diferentes modos, ante lo que el cliente ha dado libertad de elección. Finalmente, se escogió obtener los datos con el SDK de JavaScript para Facebook, pudiéndose éste integrar fácilmente en el código HTML y dando la funcionalidad suficiente al cliente y al usuario final.

4. Análisis del sistema

Esta fase está orientada a la especificación detallada del sistema, en la que se analizan los diferentes aspectos del proyecto para que éste salga adelante.

4.1. Introducción

Realizando este análisis del sistema, se pretenderá obtener una lista detallada de requisitos software que definan el sistema, basándonos en los recogidos en el punto “3. *Estudio de Viabilidad del Sistema*”. Una vez confeccionado este análisis, éste nos servirá como base para realizar posteriormente el diseño del sistema.

En esta sección, se tratará de definir el sistema, para después obtener los diferentes requisitos software, casos de uso e interfaces de usuario que permita realizar el diseño de la aplicación.

4.2. Definición del sistema

En este apartado, se definirá el sistema actual, para intentar satisfacer todas las necesidades planteadas por el cliente en el punto “3. *Estudio de Viabilidad del Sistema*”. Para ello, se determinará el alcance del sistema, se identificará el entorno tecnológico usado, se especificarán diferentes estándares y normas, y se identificarán los usuarios participantes y finales, así como la seguridad requerida en este proceso.

4.2.1. Determinación del alcance del sistema

El sistema a desarrollar es una aplicación para Facebook que permite hacer una búsqueda entre los datos de los amigos de un usuario, y opcionalmente, también entre los amigos de esos amigos del usuario, mostrando los resultados a través de un grafo y una lista de fotografías.

Entre las características fundamentales del programa, destacan las siguientes:

- La aplicación estará alojada en un servidor.
- La aplicación debe funcionar bajo la web Facebook.
- El software tendrá una interfaz intuitiva y atractiva para los usuarios.
- Se desarrollará un algoritmo que permita buscar el término o conjunto de términos deseados por el usuario entre los datos de sus amigos.
- Se generará un grafo en el que se mostrarán los resultados extraídos de la búsqueda.

4.2.2. Identificación del entorno tecnológico

Para realizar el software que se pide en este proyecto, sólo es necesario utilizar tecnología web basada en HTML y JavaScript. Para manejar esta tecnología, no es necesario un equipo demasiado potente, ya que con un simple editor de textos es suficiente. En el apartado Entorno operacional se observará en mayor detalle los elementos empleados para el desarrollo.

4.2.3. Especificación de estándares y normas

Para realizar el desarrollo software de este producto junto con su correspondiente documentación, se utilizarán los siguientes estándares y normas:

- Métrica Versión 3: Se utilizará este estándar español en diversas fases, como el Estudio de Viabilidad del Sistema, el Análisis del Sistema, y el Diseño del Sistema. Sin embargo, esta metodología será adaptada a los requerimientos del proyecto.
- UML: Este lenguaje de modelado de sistemas se utilizará para realizar el Diseño del Sistema.

4.2.3.1. Restricciones generales

A continuación, se detallarán algunas restricciones que se deben tener en cuenta a la hora de desarrollar el sistema:

- Los navegadores deben de tener habilitada la opción de ejecutar código en JavaScript para que la aplicación funcione correctamente.
- El idioma en el que debe estar escrito tanto la interfaz como la codificación es el español.
- La aplicación debe cumplir con todas las funcionalidades que se describen en los requisitos software del sistema que se detallarán más adelante.
- La aplicación debe ser intuitiva y fácil de usar.
- El tiempo de respuesta para dibujar el grafo debe ser inferior a los 5 minutos.
- El tiempo de respuesta para listar los mejores resultados debe ser inferior a los 30 segundos después de dibujar el grafo.

4.2.3.2. Supuestos y dependencias

Por un lado, suponemos que todos los usuarios participantes en la fase de análisis de este documento conocen la metodología de Métrica 3, por lo que todos ellos deberán de respetar todo lo establecido por ese estándar.

Por otro lado, la aplicación que aquí se intenta desarrollar tiene varias dependencias. Una de esas dependencias es a la política de privacidad de la compañía Facebook sobre los datos de sus usuarios; si esta política cambia, deberá de ser revisada para que la aplicación cumpla con los cambios. Otra dependencia es al nivel de privacidad establecido por cada usuario de sus datos, ya que si los tiene protegidos, no podrán ser accedidos por la aplicación.

4.2.3.3. Entorno operacional

Los lenguajes de programación a utilizar son HTML y JavaScript, por lo que no se necesita un entorno operacional especial, ya que con un simple editor de textos como el Notepad++ (Notepad++, 2011) es suficiente. También, gracias a las herramientas para administradores de los diferentes navegadores, se podrán realizar controles y pruebas de lo que sucede en el código JavaScript que generemos. Además, para la gestión de documentos, se ha utilizado la herramienta Microsoft Office 2007.

Para el desarrollo del proyecto, se ha utilizado un ordenador con las siguientes características:

- Ordenador portátil Sony Vaio:
 - Microprocesador: Intel Core i5 a 2.40 GHz.
 - Memoria RAM: 6 GB.
 - Sistema Operativo: Microsoft Windows 7.

4.2.4. Identificación de los usuarios

En este apartado se detallan los usuarios que participarán en esta fase del desarrollo. Estos participantes son los siguientes:

- Equipo de desarrollo: Grupo de personas encargadas de realizar el proyecto. Para el caso que nos ocupa, la única persona en este equipo de desarrollo será José Ángel Cuadrado Mingo, autor del proyecto, que tomará diversos roles para asegurar el desarrollo de la aplicación.
- Cliente / tutor: Persona encargada de supervisar el avance del desarrollo del proyecto. En este aspecto, se encargará también de colaborar con la obtención de requisitos del sistema. Su objetivo es que el proyecto se desarrolle con la suficiente calidad y en el tiempo y precio adecuado. En este caso, el cliente y el tutor es la misma persona, José Arturo Mora Soto.

Por otra parte, cabe destacar que entre los usuarios finales, es decir, aquellos que vayan a utilizar la aplicación, serán los encargados también de validar y aceptar dicho proyecto.

4.2.5. Estudio de la seguridad requerida en el proceso de análisis

Todos los productos generados en este proceso de análisis sólo podrán ser usados por las personas indicadas en el apartado Identificación de los usuarios.

4.3. Establecimiento de requisitos software

En esta sección, se analiza el sistema partiendo de los requisitos de usuario construyendo una descripción coherente y completa de lo que el software deberá hacer.

4.3.1. Obtención de requisitos software

A continuación, se realizará una lista detallada de los requisitos de software que se han extraído gracias a la ayuda de los requisitos del usuario descritos en el punto “3.4. Definición de requisitos de usuario” y a los casos de uso que se describirán posteriormente. Estos requisitos se pueden dividir en varios tipos:

- Requisitos funcionales: Son aquellos requisitos que especifican las funciones propias del sistema.
- Requisitos no funcionales: Son aquellos requisitos que especifican otros criterios necesarios para el correcto funcionamiento del sistema. Estos requisitos se subdividen en otra clase de requisitos:
 - Requisitos de interfaz: Son aquellos requisitos que especifican el hardware y el software necesario con los que debe interactuar el sistema.
 - Requisitos de rendimiento: Son aquellos requisitos que especifican la carga que tenga que soportar el sistema.
 - Requisitos de seguridad: Son aquellos requisitos que especifican los elementos que protegerán al software.
 - Requisitos de calidad: Son aquellos requisitos que guían a las personas implicadas en el desarrollo del software a especificar criterios para asegurar que el producto cumple ciertos niveles de calidad.

Para que los requisitos sean recogidos de la manera más clara posible y mantener una estructura de los mismos, se utilizará la plantilla mostrada en la tabla 22:

Código Requisito			
Nombre:			
Prioridad:	Alta/Media/Baja	Estabilidad:	Alta/Media/Baja
Necesidad:	Esencial/Deseable/Opcional		
Claridad:	Alta/Media/Baja	Verificabilidad:	Alta/Media/Baja
Descripción:			

Tabla 22. Plantilla de los requisitos de software

A continuación, se especificará el contenido y significado de los campos pertenecientes a la plantilla mostrada en la tabla 22.

- **Código Requisito:** Cada requisito incluirá una identificación para facilitar su traza para tareas subsiguientes. Dicho código seguirá el estándar RS-F-XX para los requisitos funcionales, RS-I-XX para los requisitos de interfaz, RS-R-XX para los requisitos de rendimiento, RS-S-XX para los requisitos de seguridad, y RS-C-XX para los requisitos de calidad, donde XX corresponde a un número ordinal.
- **Nombre:** Expresa en pocas palabras el significado del requisito.
- **Prioridad:** Indica la prioridad de desarrollo de un determinado requisito. Este campo puede tomar los valores “Alta”, “Media” y “Baja”.
- **Estabilidad:** Indica la posibilidad de que el requisito cambie a lo largo del desarrollo de la aplicación. Cuanto más alto, menor posibilidad de cambio.
- **Necesidad:** Indica el nivel de necesidad del requisito en el sistema. Este campo puede tomar los valores “Esencial”, “Deseable” u “Opcional”.
- **Claridad:** Indica la facilidad de comprensión del significado de un determinado requisito. Este campo puede tomar los valores “Alta”, “Media” o “Baja”.
- **Verificabilidad:** Indica la capacidad de verificación por parte del cliente de que se cumple un determinado requisito en el sistema final. Este campo puede tomar los valores “Alta”, “Media” o “Baja”.
- **Descripción:** Breve especificación de un determinado requisito mediante el uso de texto explicativo.

4.3.1.1. Requisitos funcionales

RS-F-01			
Nombre:	El usuario deberá estar conectado a la red social para poder ejecutar la aplicación.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Para que el usuario pueda utilizar la aplicación, deberá estar conectado a la red social para que el sistema pueda acceder a los permisos que haya concedido a la aplicación y a los datos del usuario.		

Tabla 23. RS-F-01

RS-F-02			
Nombre:	El usuario podrá utilizar la aplicación desde una página web externa a la red social.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Opcional		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	El usuario podrá hacer uso de la aplicación desde el sitio web original de la aplicación, utilizando el enlace al servidor del Laboratorio del Departamento de Informática de la Universidad Carlos III de Madrid, siempre y cuando esté conectado de forma independiente a Facebook, para cumplir con el requisito descrito en RS-F-01.		

Tabla 24. RS-F-02

RS-F-03			
Nombre:	El usuario podrá seleccionar el nivel de profundidad de la búsqueda.		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	El usuario podrá seleccionar que su búsqueda se realice entre los datos de sus amigos o también entre los datos de los amigos de sus amigos.		

Tabla 25. RS-F-03

RS-F-04			
Nombre:	El sistema realizará una búsqueda en profundidad para encontrar los resultados a la consulta planteada.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	<p>Una vez que el usuario haya seleccionado el nivel de búsqueda y el término o conjunto de términos a buscar, y ha hecho clic en el botón de buscar, la aplicación empieza a buscar entre los diferentes campos relevantes para ésta; dichos campos son:</p> <ul style="list-style-type: none"> • Nombre completo • Lugar de nacimiento y residencia • Preferencias políticas y religiosas • Estado civil • Trabajo • Educación • Gustos <p>Si la aplicación ha encontrado una similitud entre el texto buscado y los datos que se encuentran en estos campos, lo guardará como un resultado de la búsqueda. Sólo buscará en el campo “Gustos” si no ha encontrado ninguna similitud en los campos anteriores. Si el usuario desea buscar también entre los amigos de sus amigos, se buscará en los datos disponibles por la privacidad los campos indicados anteriormente.</p>		

Tabla 26. RS-F-04

RS-F-05			
Nombre:	El sistema dará una prioridad a cada nodo en función del resultado obtenido.		
Prioridad:	Alta	Estabilidad:	Media
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	<p>La aplicación dará una prioridad a cada resultado obtenido, teniendo en cuenta el campo en el que se haya encontrado la similitud. Las prioridades vienen ordenadas de la siguiente manera:</p> <ol style="list-style-type: none"> 1. Nombre completo 2. Preferencias políticas y religiosas 3. Estado civil 4. Lugar de nacimiento y residencia 5. Trabajo 6. Educación 7. Gustos 8. Similitud encontrada en un amigo de un amigo <p>Este orden ayudará de igual forma a listar los mejores resultados en la aplicación.</p>		

Tabla 27. RS-F-05

RS-F-06			
Nombre:	El sistema mostrará una imagen de espera mientras se realiza la búsqueda.		
Prioridad:	Baja	Estabilidad:	Media
Necesidad:	Opcional		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Mientras la aplicación está realizando la búsqueda, el sistema muestra en la interfaz gráfica una imagen de espera para avisar al usuario de que se está realizando la búsqueda.		

Tabla 28. RS-F-06

RS-F-07			
Nombre:	El sistema representará en un grafo los resultados obtenidos.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	El sistema mostrará en un grafo los resultados obtenidos en la búsqueda, de tal forma que el nodo principal sea el usuario que está utilizando la aplicación, los nodos que se enlazan con el principal son los amigos de éste y que tienen alguna similitud, y los nodos salientes de estos últimos son los amigos de esos amigos que tienen alguna similitud. En caso de que un amigo del usuario no tenga similitud, pero alguno de los amigos de éste sí que la tenga, se representará a ambos en el grafo tal y como se ha descrito anteriormente.		

Tabla 29. RS-F-07

RS-F-08			
Nombre:	El sistema representará cada nodo del grafo de un color distinto en función a su prioridad.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Media
Descripción:	<p>Cada nodo del grafo es un resultado obtenido en la búsqueda correspondiente a un amigo o amigo de amigo del usuario. Para encontrar la similitud, se busca entre diferentes campos tal y como se ha descrito en RS-F-05 y se da una prioridad a este usuario tal y como se describe en RS-F-06. En función a estas prioridades, los nodos estarán rellenos de los siguientes colores:</p> <ul style="list-style-type: none"> • Azul: nodo principal, representando al usuario. • Verde: nodos que representan a los amigos del usuario que tienen una prioridad de nivel 1 al 6. • Amarillo: nodos que representan a los amigos del usuario que tienen una prioridad de nivel 7. • Rojo: nodos que representan a los amigos de amigos del usuario (con prioridad de nivel 8). <p>En caso de que no se encuentre ningún resultado en los nodos de color verde o amarillo, los siguientes niveles de color bajarán como si fueran los anteriores. Es decir, si no hay nadie que represente a nodos verdes, las personas con prioridad 7 serán nodos verdes y las personas con prioridad 8 serán nodos amarillos, e igual con los nodos amarillos: si no hay ninguno de estos, las personas con prioridad 8 serán nodos amarillos.</p>		

Tabla 30. RS-F-08

RS-F-09			
Nombre:	Los resultados obtenidos se guardarán en un objeto JSON		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Media	Verificabilidad:	Media
Descripción:	<p>Los resultados que obtiene la aplicación tendrán que ser guardados en un objeto tipo JSON para poder ser utilizados por la librería Protovis para generar el grafo. Este objeto sigue la siguiente estructura:</p> <ul style="list-style-type: none"> • Resultados: <ul style="list-style-type: none"> ○ Nodos <ul style="list-style-type: none"> ▪ Nombre del nodo ▪ Grupo (prioridad) ○ Enlaces <ul style="list-style-type: none"> ▪ Origen ▪ Destino ▪ Valor (1 por defecto) 		

Tabla 31. RS-F-09

RS-F-10			
Nombre:	Los mejores resultados para listar se ordenarán en función del campo en el que encontró la similitud.		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Opcional		
Claridad:	Alta	Verificabilidad:	Media
Descripción:	<p>Los mejores resultados obtenidos en la búsqueda se corresponden a los nodos de color verde descritos en RS-F-09. Estos resultados tienen diferentes niveles de prioridad en función a lo descrito en RS-F-06. La aplicación mostrará en la interfaz gráfica estos resultados agrupados según el campo en el que se ha encontrado la similitud.</p>		

Tabla 32. RS-F-10

RS-F-11			
Nombre:	Los mejores resultados para listar se guardarán en un array multidimensional		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Opcional		
Claridad:	Alta	Verificabilidad:	Media
Descripción:	La aplicación guardará en un array los mejores resultados obtenidos correspondientes a los nodos de color verde descritos en RS-F-09. En este array, se guardará el identificador del amigo del usuario, para obtener posteriormente su imagen de perfil, y su prioridad, para poder agruparlo en función al campo en el que se encontró la similitud.		

Tabla 33. RS-F-11

RS-F-12			
Nombre:	El sistema permitirá limpiar los resultados del grafo		
Prioridad:	Media	Estabilidad:	Media
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Media
Descripción:	Una vez terminada una búsqueda, el usuario tendrá que limpiar el grafo de la pantalla para volver a realizar otra búsqueda. Para ello, bastará con que el usuario utilice un botón tipo “submit” para refrescar la pantalla.		

Tabla 34. RS-F-12

RS-F-13			
Nombre:	El sistema permitirá arrastrar un nodo		
Prioridad:	Media	Estabilidad:	Media
Necesidad:	Opcional		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	El sistema permitirá que el usuario maneje los nodos para que los utilice de la forma que crea más conveniente y mejore su experiencia con la aplicación.		

Tabla 35. RS-F-13

RS-F-14			
Nombre:	El sistema permitirá arrastrar el grafo a lo largo de la zona acondicionada para ello.		
Prioridad:	Media	Estabilidad:	Media
Necesidad:	Opcional		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	El sistema permitirá que el usuario maneje el grafo en la zona perteneciente a éste para que pueda utilizarlo de la forma que crea más conveniente y mejore su experiencia con la aplicación.		

Tabla 36. RS-F-14

RS-F-15			
Nombre:	El sistema permitirá alejar y acercar el grafo.		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	El sistema permitirá que el usuario pueda alejar o acercar el grafo para que pueda utilizarlo de la forma que crea más conveniente y mejore su experiencia con la aplicación.		

Tabla 37. RS-F-15

RS-F-16			
Nombre:	El sistema mostrará el nombre de la persona al que pertenece el nodo si el usuario posa el cursor sobre el nodo.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Cuando un usuario se pose sobre uno de los nodos, el sistema mostrará el nombre de la persona a la que pertenece ese nodo, de tal forma que pueda quedar claro que personas tienen alguna similitud con la búsqueda realizada por el usuario.		

Tabla 38. RS-F-16

RS-F-17			
Nombre:	La aplicación no será sensible a mayúsculas, pero sí a las tildes.		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Cuando un usuario vaya a buscar un término o un conjunto de ellos a través de la aplicación, no importará si lo escribe en mayúsculas o en minúsculas, puesto que el sistema no hace diferencia entre ellos. No obstante, sí que hará diferencias entre las palabras que lleven o no tilde, por lo que el usuario tendrá que tener en cuenta las tildes al realizar la búsqueda.		

Tabla 39. RS-F-17

RS-F-18			
Nombre:	El usuario podrá consultar una sección de ayuda		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	El usuario tendrá una sección de ayuda en la que podrá consultar sus dudas para aclararlas fácilmente.		

Tabla 40. RS-F-18

4.3.1.2. Requisitos de interfaz

RS-I-01			
Nombre:	La aplicación ha de verse correctamente en Internet Explorer 9, Mozilla Firefox 4 y Google Chrome 19.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Debido a que estos tres navegadores son los más importantes debido a su gran número de usuarios, se debe asegurar el correcto funcionamiento de la aplicación en estos navegadores, tanto para la versión indicada como para las sucesivas actualizaciones.		

Tabla 41. RS-I-01

RS-I-02			
Nombre:	La aplicación ha de verse correctamente con una resolución de 1280x1024.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Esta resolución se adecúa al tamaño de largo de la aplicación para su correcta visualización de una sola vez.		

Tabla 42. RS-I-02

RS-I-03			
Nombre:	El sistema utilizará las llamadas asíncronas a la API de Facebook para obtener los datos necesarios a buscar.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Para poder realizar correctamente la funcionalidad del buscador, se deben realizar varias llamadas a la API de Facebook para poder obtener los datos necesarios. Para ello, es necesario saber al menos si el usuario tiene los permisos aceptados, cuáles son sus amigos, obtener los datos de esos amigos, obtener los gustos de los amigos que no tengan ninguna similitud con los datos previos, y sí se quiere buscar entre los amigos de los amigos, se tendrá que obtener la lista de amigos de cada amigo y los datos de esos amigos de amigos.		

Tabla 43. RS-I-03

RS-I-04			
Nombre:	Los navegadores deben tener habilitada la opción de utilizar JavaScript.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Algunos navegadores tienen la opción de no ejecutar JavaScript para no instalar ni ejecutar objetos extraños durante la navegación, pero para poder ejecutar esta aplicación, será necesario que tenga habilitada la opción de utilizar y ejecutar acciones en JavaScript.		

Tabla 44. RS-I-04

RS-I-05			
Nombre:	La aplicación debe estar escrita en español.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Para mejorar la comprensión de la aplicación en el país en el que se ha desarrollado, esta herramienta tendrá que utilizar el español tanto en la interfaz como en la codificación.		

Tabla 45. RS-I-05

4.3.1.3. Requisitos de rendimiento

RS-R-01			
Nombre:	El grafo se mostrará en un tiempo máximo de 5 minutos.		
Prioridad:	Media	Estabilidad:	Media
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Media
Descripción:	Para asegurar que la búsqueda no se alarga demasiado tiempo a causa de una búsqueda en un número excesivo de personas, se limitará a que la consulta tarde como mucho 5 minutos.		

Tabla 46. RS-R-01

RS-R-02			
Nombre:	La lista de los mejores resultados se mostrará en menos de 30 segundos una vez se haya mostrado el grafo.		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Deseable		
Claridad:	Alta	Verificabilidad:	Baja
Descripción:	Una vez que se ha mostrado el grafo, el sistema no debe tardar más de 30 segundos en mostrar la lista de resultados para no hacer esperar más tiempo al usuario en mostrar el resultado final.		

Tabla 47. RS-R-02

4.3.1.4. Requisitos de seguridad

RS-S-01			
Nombre:	La aplicación podrá ejecutarse desde una conexión segura.		
Prioridad:	Media	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Para que todos los usuarios puedan manejar la aplicación de forma segura, se realizará una vía para que dicha herramienta pueda ser usada, por los usuarios que lo deseen, desde una conexión segura.		

Tabla 48. RS-S-01

RS-S-02			
Nombre:	El usuario deberá conceder una lista de permisos a la aplicación para poder ejecutarse.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	<p>Para que el usuario pueda utilizar la aplicación, deberá conceder la lista de permisos que se muestra a continuación:</p> <ul style="list-style-type: none"> • User_education_history • User_hometown • User_likes • User_location • User_relationships • User_religion_politics • User_work_history • Friends_education_history • Friends_hometown • Friends_likes • Friends_location • Friends_relationships • Friends_religion_politics • Friends_work_history <p>De esta forma, la aplicación podrá conocer los diferentes aspectos de búsqueda entre los que buscar en los amigos del usuario con la información solicitada por este último.</p>		

Tabla 49. RS-S-02

4.3.1.5. Requisitos de calidad

RS-C-01			
Nombre:	El sistema debe superar con éxito las pruebas que se realicen para dar el sistema como aceptado.		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Para que la aplicación tenga los suficientes niveles de calidad, deberá de pasar una serie de pruebas que demuestren que todo funcione correctamente.		

Tabla 50. RS-C-01

4.3.2. Especificación de casos de uso

Para ayudar a la tarea de obtener los requisitos software del sistema, se utilizará las técnicas de los casos de uso. De esta forma, se podrán describir los pasos a seguir en cada una de las posibles acciones llevadas a cabo por el usuario y/o el sistema.

Para mostrar la información de los casos de uso, seguiremos la plantilla mostrada en la tabla 51.

Identificador	Nombre
Actores	
Objetivo	
Precondiciones	
Post-condiciones	
Escenario	
Condiciones de fallo	
Casos de uso relacionados	

Tabla 51. Plantilla de los casos de uso

Seguidamente, se especificará el contenido y significado de los campos pertenecientes a la plantilla mostrada en la tabla 51.

- **Identificador:** Identifica al caso de uso. Se define como CU-XX, donde XX es un número ordinal.
- **Nombre:** Nombre que identificará el caso de uso.
- **Actores:** Los posibles roles de usuario que pueden intervenir en el caso de uso.

- **Objetivo:** La descripción de la finalidad que pretende el actor con la realización de este caso de uso y breve explicación del proceso llevado a cabo en el caso de uso.
- **Pre-condiciones:** Son aquellos hechos que deben ser previamente ciertos para poder llevar a cabo el caso de uso.
- **Post-condiciones:** Son aquellos hechos que la ejecución del caso de uso hace verdaderos o ciertos.
- **Escenario:** Descripción esquemática de las fases que componen el caso de uso.
- **Condiciones de fallo:** Describe posibles errores y las respuestas del sistema ante los mismos.

A continuación, se muestra la imagen 14 en la que aparece el sistema con los diferentes casos de uso, así como la descripción general de cada uno de ellos siguiendo la plantilla anterior.

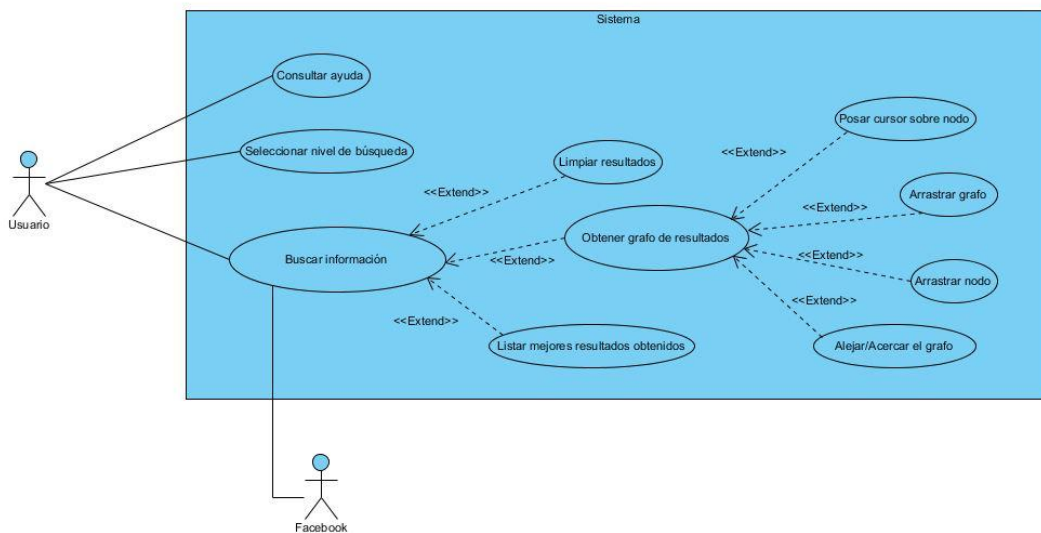


Imagen 14. Casos de uso del sistema

CU-01	Seleccionar nivel de búsqueda
Actores	Usuario
Objetivo	El usuario podrá escoger la profundidad del grafo en función de si quiere buscar la información entre sus amigos o también entre los amigos de sus amigos.
Precondiciones	<ul style="list-style-type: none"> Los usuarios a buscar deben tener pública su información para poder obtener datos de ellos.
Post-condiciones	---
Escenario	<ol style="list-style-type: none"> El usuario selecciona una opción de nivel de búsqueda. El sistema guarda la opción escogida.
Condiciones de fallo	<ul style="list-style-type: none"> El sistema no guarda correctamente la opción seleccionada por el usuario.
Casos de uso relacionados	---

Tabla 52. CU-01

CU-02	Buscar información
Actores	Usuario, Facebook
Objetivo	El usuario quiere buscar una cierta información entre los datos de sus amigos almacenados en la base de datos de Facebook, y el sistema le devuelve un grafo con los resultados obtenidos.
Precondiciones	<ul style="list-style-type: none"> Los usuarios a buscar deben tener pública su información para poder obtener datos de ellos. El usuario que utiliza la aplicación debe haber aceptado los permisos necesarios para el funcionamiento de la aplicación. El usuario debe haber limpiado resultados anteriores en caso de que haya.
Post-condiciones	<ul style="list-style-type: none"> El sistema muestra un grafo con la relación de los resultados obtenidos, así como una lista de los usuarios resultantes.
Escenario	<ol style="list-style-type: none"> El usuario hace clic en el botón “Limpiar” en el caso en el que esté disponible. El usuario escribe en el cuadro de texto la información que quiere buscar. El usuario hace clic en el botón “Buscar”. El sistema inhabilita el botón de “Buscar” y habilita el botón de “Limpiar”. El sistema realiza la búsqueda en función a los parámetros especificados. El sistema muestra los resultados obtenidos.
Condiciones de fallo	<ul style="list-style-type: none"> El sistema no puede obtener resultados ya que el usuario no tiene permisos suficientes. El usuario no introdujo nada en el campo de texto y pulsó el botón. El sistema no puede obtener resultados ya que los amigos de los usuarios tienen una alta protección de su privacidad. El sistema no inhabilita correctamente el botón “Buscar”. El sistema no habilita correctamente el botón “Limpiar”.
Casos de uso relacionados	<ul style="list-style-type: none"> CU-04 (caso de uso hijo) CU-05 (caso de uso hijo) CU-06 (caso de uso hijo)

Tabla 53. CU-02

CU-03	Consultar ayuda
Actores	Usuario
Objetivo	El usuario quiere leer una pequeña explicación de la herramienta antes de utilizarla.
Precondiciones	---
Post-condiciones	---
Escenario	<ol style="list-style-type: none"> 1. El usuario hace clic en el botón “Ayuda”. 2. El sistema muestra una alerta con la ayuda.
Condiciones de fallo	---
Casos de uso relacionados	---

Tabla 54. CU-03

CU-04	Limpiar resultados
Actores	Usuario
Objetivo	El usuario limpia los resultados de una búsqueda anterior antes de realizar una nueva consulta.
Precondiciones	<ul style="list-style-type: none"> • El usuario debe haber realizado una consulta de forma correcta anteriormente.
Post-condiciones	<ul style="list-style-type: none"> • El sistema debe dejar la aplicación como en su estado inicial.
Escenario	<ol style="list-style-type: none"> 1. El usuario hace una búsqueda. 2. El sistema muestra los resultados obtenidos. 3. El usuario hace clic en el botón “Limpiar”. 4. El sistema refresca la aplicación y la deja en su estado inicial.
Condiciones de fallo	<ul style="list-style-type: none"> • El sistema no limpia correctamente los resultados. • El sistema no permite volver a realizar una búsqueda.
Casos de uso relacionados	<ul style="list-style-type: none"> • CU-02 (caso de uso padre)

Tabla 55. CU-04

CU-05	
Listar mejores resultados obtenidos	
Actores	Usuario
Objetivo	El sistema muestra los mejores resultados obtenidos de la búsqueda en función a las diferentes prioridades sacadas en ella, y mostrándolo en forma de lista de imágenes, siendo éstas las imágenes de perfil de cada usuario.
Precondiciones	<ul style="list-style-type: none"> El usuario debe de haber realizado una búsqueda y que ésta haya terminado correctamente. Los usuarios a buscar deben tener pública su información para poder obtener datos de ellos.
Post-condiciones	---
Escenario	<ol style="list-style-type: none"> El usuario hace una búsqueda. El sistema obtiene los resultados. El sistema muestra las fotografías de perfil de los usuarios con mejor prioridad, situando cada una en su categoría correspondiente en función de la prioridad.
Condiciones de fallo	<ul style="list-style-type: none"> El sistema no puede mostrar la información ya que los amigos de los usuarios tienen una alta protección de su privacidad. El sistema no puede mostrar la información ya que la búsqueda previa no ha sido correcta. El sistema no puede listar los mejores resultados ya que éstos no se han almacenado correctamente. El sistema no puede mostrar las fotografías porque ha sobrepasado el tiempo límite.
Casos de uso relacionados	<ul style="list-style-type: none"> CU-02 (caso de uso padre)

Tabla 56. CU-05

CU-06	
Obtener grafo de resultados	
Actores	Usuario
Objetivo	El sistema muestra un grafo manejable con los resultados obtenidos de la búsqueda, mostrando una escala de prioridades en función del color del nodo.
Precondiciones	<ul style="list-style-type: none"> • El usuario debe de haber realizado una búsqueda y que ésta haya terminado correctamente. • Los usuarios a buscar deben tener pública su información para poder obtener datos de ellos.
Post-condiciones	<ul style="list-style-type: none"> • El grafo debe poder manejarse.
Escenario	<ol style="list-style-type: none"> 1. El usuario hace una búsqueda. 2. El sistema obtiene los resultados. 3. El sistema muestra el grafo representando a los usuarios en nodos y sus enlaces, y pudiendo manejar estos al antojo del usuario.
Condiciones de fallo	<ul style="list-style-type: none"> • El sistema no puede mostrar la información ya que los amigos de los usuarios tienen una alta protección de su privacidad. • El sistema no puede mostrar la información ya que la búsqueda previa no ha sido correcta. • El sistema no puede mostrar el grafo ya que los resultados no se han almacenado correctamente. • El sistema no puede mostrar el grafo correctamente. • El grafo no puede manejarse.
Casos de uso relacionados	<ul style="list-style-type: none"> • CU-02 (caso de uso padre) • CU-07 (caso de uso hijo) • CU-08 (caso de uso hijo) • CU-09 (caso de uso hijo) • CU-10 (caso de uso hijo)

Tabla 57. CU-06

CU-07 Posar cursor sobre nodo	
Actores	Usuario
Objetivo	El usuario, al posar el cursor del ratón sobre cualquier nodo del grafo, el sistema muestra el nombre del usuario a quién corresponde dicho nodo.
Precondiciones	<ul style="list-style-type: none"> El usuario debe de haber realizado una búsqueda y que ésta haya terminado correctamente. Los usuarios a buscar deben tener pública su información para poder obtener datos de ellos. El grafo debe poder manejarse.
Post-condiciones	---
Escenario	<ol style="list-style-type: none"> El usuario hace una búsqueda. El sistema obtiene los resultados y muestra el grafo. El usuario mueve el cursor del ratón hacia un nodo cualquiera del grafo El sistema muestra el nombre del usuario al que pertenece el nodo.
Condiciones de fallo	<ul style="list-style-type: none"> El sistema no puede mostrar la información ya que los amigos de los usuarios tienen una alta protección de su privacidad. El sistema no puede mostrar la información ya que la búsqueda previa no ha sido correcta. El sistema no puede mostrar el grafo ya que los resultados no se han almacenado correctamente. El sistema no puede mostrar el grafo correctamente. El grafo no puede manejarse.
Casos de uso relacionados	<ul style="list-style-type: none"> CU-06 (caso de uso padre)

Tabla 58. CU-07

CU-08 Arrastrar grafo	
Actores	Usuario
Objetivo	El usuario, al mantener pulsado el ratón sobre la zona del grafo, puede arrastrarlo hacia cualquier zona de la delimitada para él.
Precondiciones	<ul style="list-style-type: none"> El usuario debe de haber realizado una búsqueda y que ésta haya terminado correctamente. El grafo debe poder manejarse.
Post-condiciones	---
Escenario	<ol style="list-style-type: none"> El usuario hace una búsqueda. El sistema obtiene los resultados y muestra el grafo. El usuario pulsa el botón sobre la zona del grafo y lo arrastra. El sistema mueve el grafo según lo hace el usuario.
Condiciones de fallo	<ul style="list-style-type: none"> El sistema no puede mostrar la información ya que los amigos de los usuarios tienen una alta protección de su privacidad. El sistema no puede mostrar la información ya que la búsqueda previa no ha sido correcta. El sistema no puede mostrar el grafo ya que los resultados no se han almacenado correctamente. El sistema no puede mostrar el grafo correctamente. El grafo no puede manejarse.
Casos de uso relacionados	<ul style="list-style-type: none"> CU-06 (caso de uso padre)

Tabla 59. CU-08

CU-09 Arrastrar nodo	
Actores	Usuario
Objetivo	El usuario, al mantener pulsado el ratón sobre un nodo del grafo, puede arrastrar dicho nodo hacia cualquier zona de la delimitada para el grafo.
Precondiciones	<ul style="list-style-type: none"> El usuario debe de haber realizado una búsqueda y que ésta haya terminado correctamente. El grafo debe poder manejarse.
Post-condiciones	<ul style="list-style-type: none"> El nodo no puede situarse encima de otro nodo.
Escenario	<ol style="list-style-type: none"> El usuario hace una búsqueda. El sistema obtiene los resultados y muestra el grafo. El usuario pulsa el botón sobre el nodo del grafo y lo arrastra. El sistema mueve el nodo según lo hace el usuario.
Condiciones de fallo	<ul style="list-style-type: none"> El sistema no puede mostrar la información ya que los amigos de los usuarios tienen una alta protección de su privacidad. El sistema no puede mostrar la información ya que la búsqueda previa no ha sido correcta. El sistema no puede mostrar el grafo ya que los resultados no se han almacenado correctamente. El sistema no puede mostrar el grafo correctamente. El grafo no puede manejarse.
Casos de uso relacionados	<ul style="list-style-type: none"> CU-06 (caso de uso padre)

Tabla 60. CU-09

CU-10 Alejar/Acercar el grafo	
Actores	Usuario
Objetivo	El usuario puede alejar o acercar el grafo utilizando la rueda del ratón para ver un enfoque global del grafo o una zona concreta de él.
Precondiciones	<ul style="list-style-type: none"> El usuario debe de haber realizado una búsqueda y que ésta haya terminado correctamente. El grafo debe poder manejarse.
Post-condiciones	---
Escenario	<ol style="list-style-type: none"> El usuario hace una búsqueda. El sistema obtiene los resultados y muestra el grafo. El usuario mueve la rueda del ratón hacia abajo/arriba. El sistema aleja/acerca el grafo según la acción del usuario.
Condiciones de fallo	<ul style="list-style-type: none"> El sistema no puede mostrar la información ya que los amigos de los usuarios tienen una alta protección de su privacidad. El sistema no puede mostrar la información ya que la búsqueda previa no ha sido correcta. El sistema no puede mostrar el grafo ya que los resultados no se han almacenado correctamente. El sistema no puede mostrar el grafo correctamente. El grafo no puede manejarse.
Casos de uso relacionados	<ul style="list-style-type: none"> CU-06 (caso de uso padre)

Tabla 61. CU-10

4.4. Análisis de la estructura del sistema

En este apartado, se va a tratar la estructura general del sistema una vez analizados los diferentes casos de uso y los requisitos software descritos anteriormente.

En nuestro sistema, se pueden identificar dos componentes principales: por un lado la aplicación, con sus diferentes opciones a escoger por el usuario, y la mecánica interna de la aplicación que llama a los métodos de Facebook para obtener los datos con los que se hace la búsqueda solicitada por el usuario.

En el diagrama de actividades de la imagen 15, se puede ver el esquema que sigue cualquier acción que pueda tomar el usuario en la aplicación.

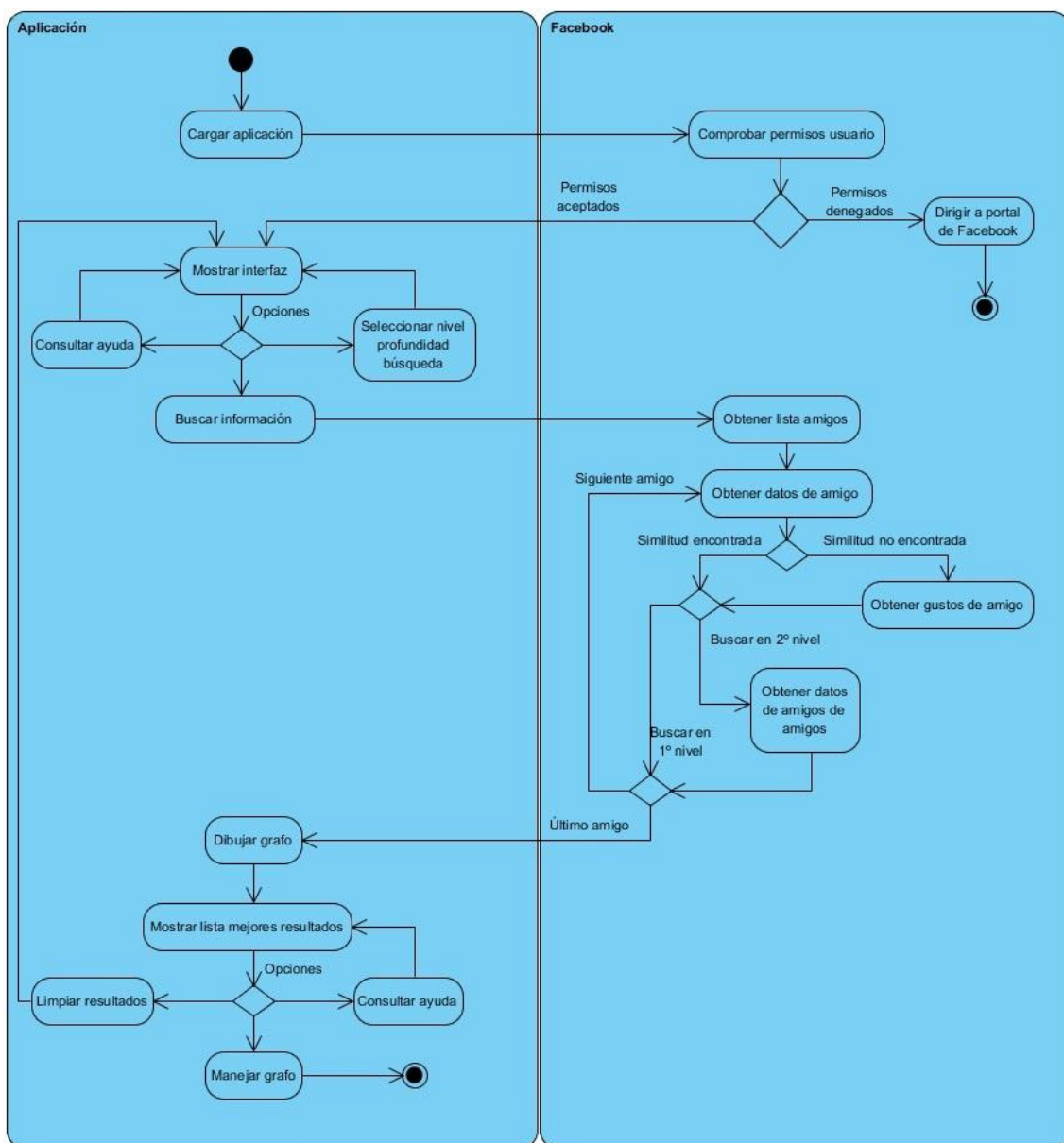


Imagen 15. Estructura de la aplicación

Como se puede observar en la imagen 15, la aplicación empieza cargándose y consultando a Facebook si ha aceptado los permisos –en caso de que no lo haya hecho, le redirige a la página principal de Facebook-. Una vez cargada la aplicación, se muestra la interfaz gráfica de ésta, y aquí el usuario tiene varias opciones: puede consultar la ayuda, puede cambiar el nivel de profundidad de la búsqueda, o puede directamente buscar la información que desea.

Cuando el usuario busca una cierta información, obtiene la lista de amigos del usuario, y por cada amigo, se obtienen los datos de éste. En caso de no encontrar una similitud con la búsqueda, se pasa a obtener los gustos del amigo. Una vez buscados estos datos, si el usuario seleccionó buscar en un segundo nivel de profundidad (amigos de amigos), se obtienen los datos de estas personas. Este proceso se repite por cada amigo del usuario, y una vez que se haya terminado esta lista, la aplicación dibuja el grafo con los resultados obtenidos, y muestra a continuación la lista con los mejores resultados.

Llegados a este punto, el usuario puede volver a consultar la ayuda, puede manejar el grafo a su antojo (moviendo los nodos, el grafo, alejándolo o acercándolo y consultar el nombre de cada nodo), o puede limpiar la interfaz y volver al inicio de la aplicación. Este proceso se repite hasta que el usuario quiera dejar de utilizar la aplicación.

4.5. Definición de las interfaces de usuario

En este apartado se definirán diferentes características de las interfaces de usuario, siendo éste uno de los puntos más importantes de cualquier aplicación, al ser el medio de comunicación entre el usuario y el sistema.

4.5.1. Especificación de principios generales de la interfaz

La interfaz que se desarrolle debe ser intuitiva y fácil de usar para todos los usuarios, y de esta manera, facilitar el uso y mejorar la experiencia de todas las personas que vayan a utilizar la aplicación.

La aplicación en sí sólo contiene una única interfaz en la que se mostrará el menú de acciones en las que podrá consultar la ayuda, seleccionar el nivel de búsqueda, y el campo de texto en el que buscar los contenidos deseados.

Justo en la parte inferior a este menú de acciones, se concentra la zona del grafo y la zona en la que se listan a los mejores resultados obtenidos.

4.5.2. Identificación de perfiles y diálogos

Para esta aplicación sólo existirá un tipo de usuario, que será un usuario de la red social Facebook, que esté conectado a dicha red social, y que inicie la aplicación.

4.5.3. Especificación de formatos individuales de la interfaz de la pantalla

En este apartado, se mostrará un boceto de la interfaz gráfica que se realizará de la aplicación, en la que se espera tener una visión clara de los diferentes elementos que se desarrollarán en ella, ayudado con una descripción de cada parte.

La interfaz que se plantea diseñar es la que se presenta en la imagen 16.

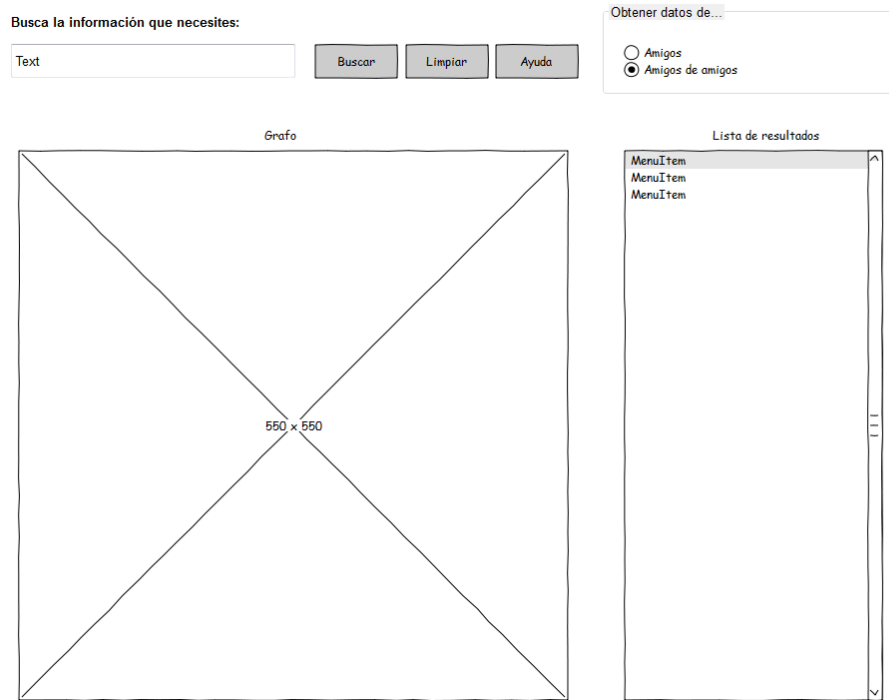


Imagen 16. Boceto de la interfaz gráfica

En la imagen 16 se pueden apreciar los diferentes elementos:

- Área de texto: En esta área, el usuario podrá escribir la información que desea buscar.
- Botón “Buscar”: Con este botón, el usuario podrá empezar a buscar la información que ha escrito en el área de texto.
- Botón “Limpiar”: Con este botón, el usuario podrá limpiar los resultados dibujados en pantalla para empezar una nueva búsqueda.
- Botón “Ayuda”: Con este botón, el usuario podrá consultar la ayuda del programa para resolver cualquier duda que tuviera.
- Elementos botón tipo radiales: Con este conjunto de elementos, el usuario podrá seleccionar el nivel de profundidad de la búsqueda, pudiendo escoger entre buscar los datos de sus amigos o también de los amigos de sus amigos.
- Imagen: En esta imagen se representará el dibujo del grafo manejable con los resultados obtenidos.
- Lista de elementos: En esta lista, se mostrarán los mejores resultados obtenidos durante la búsqueda.

4.6. Análisis de consistencia y especificación de requisitos

En este apartado, se asegurará la consistencia entre los diferentes modelos generados para que de esta forma se demuestre que haya un cierto nivel de calidad en lo expuesto hasta ahora.

4.6.1. Verificación y análisis de consistencia entre modelos

Debido a que se generan diferentes modelos de datos, es necesario comprobar que no hay incoherencias entre ellas. De esta forma, debemos comprobar que los diferentes modelos que se exponen a continuación cumplen con las siguientes normas:

- Consistencia entre requisitos de software y requisitos de usuario:
 - Todo requisito de software debe tener al menos un requisito de usuario asignado.
 - Todo requisito de usuario debe tener al menos un requisito de software asignado.
- Consistencia entre requisitos de software y casos de uso:
 - Todo requisito de software debe tener al menos un caso de uso asignado.
 - Todo caso de uso debe tener al menos un requisito de software asignado.

Para poder realizar estas comprobaciones, se utilizarán matrices de trazabilidad con los elementos que se acaban de mencionar.

4.6.1.1. Matriz de trazabilidad RS-RU

En esta matriz (tabla 62), se muestran las correspondencias entre los requisitos de software de este análisis y los requisitos de usuario del punto “3.4. Definición de requisitos de usuario”.

	RU-C-01	RU-C-02	RU-C-03	RU-C-04	RU-C-05	RU-C-06	RU-C-07	RU-C-08	RU-C-09	RU-C-10	RU-R-01	RU-R-02	RU-R-03	RU-R-04	RU-R-05	RU-R-06
RS-F-01	X															
RS-F-02	X															
RS-F-03			X													
RS-F-04		X														
RS-F-05					X											
RS-F-06				X												
RS-F-07						X										
RS-F-08						X										
RS-F-09						X										
RS-F-10							X									
RS-F-11							X									
RS-F-12		X														
RS-F-13								X								
RS-F-14								X								
RS-F-15								X								
RS-F-16								X								
RS-F-17		X														
RS-F-18									X							
RS-I-01													X			
RS-I-02														X		
RS-I-03	X	X			X						X					
RS-I-04													X			
RS-I-05															X	
RS-R-01																X
RS-R-02																X
RS-S-01										X						
RS-S-02	X											X				
RSC-01		X	X	X		X	X	X	X	X			X	X		

Tabla 62. Matriz de trazabilidad RS-RU

4.6.1.2. Matriz de trazabilidad RS-CU

En esta matriz (tabla 63), se muestran las correspondencias entre los requisitos de software y los casos de uso realizados durante este análisis.

	CU-01	CU-02	CU-03	CU-04	CU-05	CU-06	CU-07	CU-08	CU-09	CU-10
RS-F-01	X	X	X							
RS-F-02	X	X	X							
RS-F-03	X									
RS-F-04		X								
RS-F-05					X	X				
RS-F-06		X								
RS-F-07						X				
RS-F-08						X				
RS-F-09						X				
RS-F-10					X					
RS-F-11					X					
RS-F-12				X						
RS-F-13									X	
RS-F-14								X		
RS-F-15										X
RS-F-16							X			
RS-F-17		X								
RS-F-18			X							
RS-I-01	X	X	X							
RS-I-02	X	X	X							
RS-I-03		X			X	X				
RS-I-04					X	X	X	X	X	X
RS-I-05	X		X		X					
RS-R-01						X				
RS-R-02					X					
RS-S-01	X	X	X							
RS-S-02	X	X								
RS-C-01	X	X	X	X	X	X	X	X	X	X

Tabla 63. Matriz de trazabilidad RS-CU

4.6.2. Validación de los modelos

Una vez realizadas las matrices de trazabilidad, se puede verificar que los distintos elementos que aquí se han analizado son consistentes y coherentes con lo descrito en ellos.

5. Diseño del sistema

En este apartado se trata de resolver el problema planteado en el punto “4. *Análisis del sistema*”, definiendo la arquitectura del sistema y del entorno tecnológico que le da soporte, así como de los diferentes componentes de la aplicación.

5.1. Introducción

Con la definición de la arquitectura del sistema, del entorno tecnológico y de las diferentes componentes de la aplicación, se generan todas las especificaciones de construcción necesarias para implementar el sistema. Un buen resultado obtenido en la etapa del diseño facilita en gran medida la implementación del sistema, ya que se proporciona la estructura básica del sistema y la forma en que se relacionan los componentes del sistema.

En este Diseño del sistema se realizará la arquitectura del sistema que mejor convenga al sistema que aquí se desarrolla. También se determinarán los mecanismos que servirán de guía para el diseño, para posteriormente, identificar el diseño físico del sistema.

Para realizar estos pasos, será necesario contar con el entorno tecnológico adecuado para elegir el correcto entorno de desarrollo para la implementación.

5.2. Definición de la arquitectura del sistema

En la arquitectura desarrollada en este sistema se utilizan los diferentes componentes existentes para que interactúen entre ellos; esta arquitectura, por tanto, no es ninguna que sea conocida o esté estandarizada.

Debido a la sencillez del software utilizado y a la inexistencia de subsistemas, no se realizarán identificación de subsistemas ni especificación de excepciones.

5.2.1. Definición de niveles de arquitectura

El sistema contará con dos módulos: uno será el propio sistema de la aplicación, mientras que el segundo será la base de datos de los servidores de Facebook. Estos módulos deberán de conectarse mediante llamadas del primero al segundo, y con la respuesta de éste último a la aplicación. En la imagen 17 se pueden ver estos módulos y su conexión.

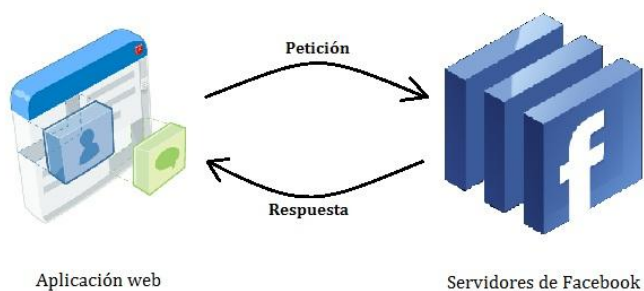


Imagen 17. Definición de niveles de arquitectura

5.2.2. Especificación de estándares y normas de diseño y construcción

Para que la tarea de la codificación de la aplicación sea más fácil para todas las personas involucradas en el desarrollo de ésta, se creará un estándar que permita unificar todos los criterios y facilitar la tarea. A continuación, se detallarán los aspectos más relevantes a tener en cuenta.

- **Nombre de ficheros:** Los diferentes ficheros que forman el código fuente deberán ser representativos de la funcionalidad que contiene cada uno. El nombre de estos ficheros empezará por letra minúscula; en caso de que contenga dos o más palabras, se concatenarán las palabras, siendo la primera letra de las palabras concatenadas mayúscula.

- Idioma: El lenguaje para la documentación del código estará obligatoriamente en castellano, mientras que el propio código no tiene la obligatoriedad de estar escrito en castellano.
- Constantes: En caso de utilizar constantes, éstas se escribirán con todas sus letras en mayúsculas.
- Variables: Las variables comenzarán por letra minúscula; si la variable está formada por más de una palabra, éstas se concatenarán, y a partir de la segunda palabra, la primera letra se escribirá en mayúscula.
- Funciones: Las funciones comenzarán por letra minúscula; si la función está formada por más de una palabra, éstas se concatenarán, y a partir de la segunda palabra, la primera letra se escribirá en mayúscula.
- Comentarios: Todos los ficheros de código deberán tener al inicio de éste un comentario mostrando el nombre del fichero, su descripción, el autor y la fecha. Las funciones tendrán un comentario explicativo de su uso justo antes al inicio de éstas. El resto de código deberá ser comentado en bloques de tal forma que se puedan entender todas las líneas de código.

5.2.3. Especificación del entorno tecnológico

Para analizar la especificación del entorno tecnológico, se remite al lector al punto “4.2.3.3. Entorno operacional”.

5.2.4. Especificación de requisitos de seguridad y operación

Para analizar la especificación de los requisitos de seguridad, se remite al lector al punto “4.3.2.4. Requisitos de seguridad”.

5.2.5. Estudio de la seguridad requerida en el proceso de diseño del sistema

En la aplicación, se manejan los diferentes datos de los usuarios de Facebook que utilizan la aplicación y de las personas amigas de dichos usuarios. Esta información es sensible, ya que requiere obtener una serie de permisos por parte de los usuarios, además de tener una cierta dependencia con el nivel de privacidad que tengan los usuarios.

Debido a esto, aparte de lo tratado en el punto “4.3.2.4. Requisitos de seguridad”, se recomienda encarecidamente tratar los diferentes aspectos en el punto “5. Diseño del sistema” con la mayor seguridad posible, transmitiendo esta información únicamente a los miembros interesados en su desarrollo.

5.3. Diseño de la arquitectura de soporte

En este apartado, se presentan los diferentes componentes de la arquitectura que forman el sistema. Este sistema cuenta con dos módulos: el módulo principal de la aplicación, y la base de datos de los servidores de Facebook. Estos dos módulos están interconectados para poder realizar las funcionalidades pedidas a la aplicación. En la imagen 18 se puede ver la relación entre ambos componentes.



Imagen 18. Componentes del sistema

Para mostrar la información de los componentes, usaremos la plantilla usada en la tabla 64.

Identificador	Nombre
Propósito	
Función	
Dependencias	
Recursos	

Tabla 64. Plantilla de componentes

A continuación, se muestran las diferentes características que se definirán sobre estos componentes.

- **Identificador:** Identifica el componente. Para ello, seguirá el patrón C-XXX, donde XXX es un número ordinal.
- **Nombre:** Nombre del componente.
- **Propósito:** Fin del componente.
- **Función:** De qué se encarga el componente.
- **Dependencias:** Componente que depende de otro para realizar sus funciones.
- **Recursos:** Hace referencia a los elementos de tipo hardware que el componente puede necesitar para realizar su función.

C-001**Aplicación**

Propósito	Permite desarrollar la búsqueda de un dato sobre los amigos y amigos de amigos de un usuario de Facebook.
Función	Solicita los datos de los amigos y amigos de amigos a Facebook para poder comparar estos con el dato escrito por el usuario.
Dependencias	Facebook
Recursos	Conexión a Internet

Tabla 65. C-001

C-002**Facebook**

Propósito	Componente externo que suministra los datos pedidos al sistema.
Función	Devuelve a la aplicación los datos necesarios para realizar la búsqueda pedida por el usuario.
Dependencias	Ninguna.
Recursos	Conexión a Internet

Tabla 66. C-002

5.4. Diseño físico de datos

Todos los datos necesarios para realizar la búsqueda pedida por el cliente se encuentran en la base de datos de Facebook. Dichos datos pueden obtenerse de diversas maneras, con diferentes entornos de programación y diferentes métodos en función de lo que sea necesario obtener.

Todos estos datos almacenados en Facebook siguen una estructura con diferentes tablas y campos. Estos datos pueden ser vistos gracias a la herramienta Graph API Explorer (Facebook, 2012) que describimos en el punto “2.5.1. Desarrollo de aplicaciones en Facebook”.

Observando las diferentes partes de esta herramienta, podemos ver que los datos que utiliza Facebook son objetos JSON, en los que los diferentes campos están compuestos de otros que lo definen en mayor grado, contando con varios objetos y arrays de objetos dentro de otros en el que se identifican y nombran los diferentes elementos.

En la imagen 7, se muestra un ejemplo de los datos que se pueden obtener en una de las tablas –en este caso, la tabla de datos de un usuario-, y como a su derecha, se muestran las explicaciones de los diferentes campos y las conexiones con otras tablas relacionadas con ese usuario en la que se encuentran otros tipos de datos. En la imagen 19 se puede observar otra de esas tablas relacionadas al usuario a través de la URL, y como ésta contiene otra serie de datos.

GET Enviar

```
{
  "data": [
    {
      "name": "Dianova España",
      "category": "Non-profit organization",
      "id": "361027433826",
      "created_time": "2012-05-30T13:47:52+0000"
    },
    {
      "name": "Goth Girls",
      "category": "Community",
      "id": "141309032579417",
      "created_time": "2012-05-28T16:16:52+0000"
    },
    {
      "name": "En España salimos de la universidad pre-parados.",
      "category": "Community",
      "id": "137585732982677",
      "created_time": "2012-05-13T20:45:01+0000"
    },
    {
      "name": "Centro de Estética Avanzada Sonia Cruz",
      "category": "Spas/beauty/personal care",
      "id": "232740570144087",
      "created_time": "2012-05-12T16:33:53+0000"
    },
    {
      "name": "Risto Mejide",
      "category": "Monarch",
      "id": "147311595390660",
      "created_time": "2012-05-09T04:14:40+0000"
    },
    {
      "name": "Mi vida",
      "category": "Entertainer",
      "id": "362157380494235",
      "created_time": "2012-05-03T21:18:49+0000"
    }
  ]
}
```

Back to Parent

Conexiones

accounts activities adaccounts albums
 apprequests books checkins events family
 feed friendlists friendrequests friends games
 groups home inbox interests likes links
 movies music mutualfriends notes notifications
 outbox payments permissions photos picture
 posts scores statuses tagged television
 updates videos

Fields

Imagen 19. Ejemplo de datos en Facebook

Una vez conocidos como es la estructura de los datos de Facebook, se puede hablar acerca de la comunicación de estos datos entre los diferentes componentes del sistema. Esta comunicación sigue una serie de pasos que se describen a continuación, y se muestran también en la imagen 20.

1. El usuario carga la página principal de Facebook.
2. El servidor de Facebook le devuelve al ordenador la página y se la muestra en el ordenador.
3. El usuario carga la aplicación.
4. El servidor de la aplicación le pide al servidor de Facebook los permisos que tiene el usuario sobre la aplicación para saber si los ha aceptado o no.
5. El servidor de Facebook le responde con los permisos que tiene el usuario.
6. El servidor de la aplicación le solicita al servidor de Facebook la inicialización del SDK de JavaScript.
7. El servidor de Facebook le responde a la aplicación con los datos de conexión entre ellos.
8. El servidor de la aplicación le devuelve al ordenador del usuario la aplicación web cargada en Facebook.
9. El usuario, al realizar una búsqueda, hace que la aplicación realice una petición al servidor de Facebook con los datos de sus amigos.
10. El servidor de Facebook le responde a la aplicación con un objeto JSON con los datos de cada amigo.

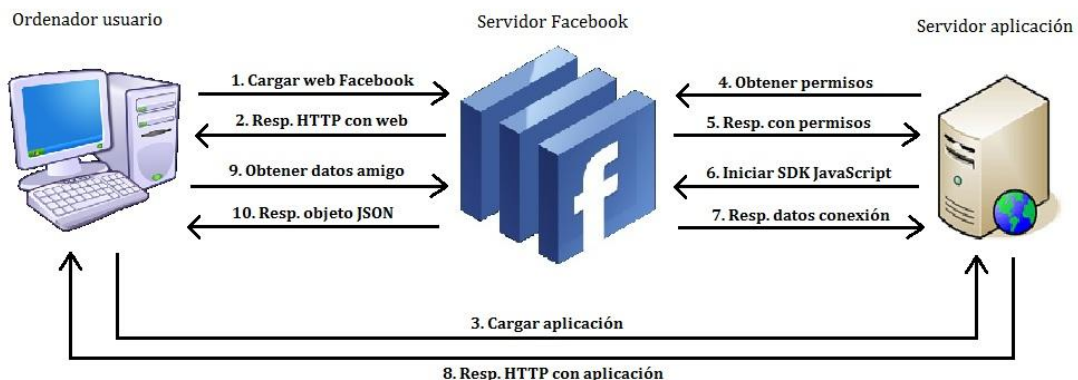


Imagen 20. Comunicación de datos

5.5. Establecimiento de requisitos de implantación

En este apartado, se especifican los requisitos de implantación del proyecto. Estos requisitos son necesarios para preparar la implantación del sistema de forma adecuada para su posterior utilización.

5.5.1. Especificación de requisitos de implantación

Utilizando la plantilla descrita en la tabla 22, a continuación se recogen los diferentes requisitos de implantación, utilizando un identificador único cuyo formato es RI-XX, donde XX es un número ordinal.

RI-01			
Nombre:	Disponer de un ordenador con conexión a Internet		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Para poder acceder a la aplicación, será necesario tener una conexión a Internet, ya que sin ella no se podrá acceder a Facebook ni a su servidor de datos.		

Tabla 67. RI-01

RI-02			
Nombre:	Disponer de un ordenador con un navegador con JavaScript		
Prioridad:	Alta	Estabilidad:	Alta
Necesidad:	Esencial		
Claridad:	Alta	Verificabilidad:	Alta
Descripción:	Para poder acceder a la aplicación, será necesario contar un navegador de Internet (preferiblemente los establecidos en el requisito RS-I-01 – tabla 41-) que permita el uso de JavaScript en los sitios web (descrito en el requisito RS-I-04 –tabla 44-).		

Tabla 68. RI-02

6. Implementación del sistema

Durante esta fase, se ha desarrollado el código de la aplicación. Como ya se ha dicho en apartados anteriores, el código desarrollado usa tecnología web basada en HTML y JavaScript, usando además el SDK de JavaScript para Facebook para obtener los datos necesarios.

En este apartado se describe el funcionamiento de la interfaz que compone el sistema, el algoritmo de búsqueda usado, y la muestra de los resultados en pantalla a través del grafo y de la lista de mejores resultados.

6.1. Interfaz de la aplicación

La interfaz que se ha desarrollado para la aplicación ha sido similar a la que se detalló en el punto “4.5.3. Especificación de formatos individuales de la interfaz de la pantalla”.

Antes de mostrar la interfaz, al intentar entrar en la aplicación, se consulta que el usuario ha aceptado los permisos para utilizar la aplicación. En caso de que no la haya aceptado aún, le saldrá un mensaje en pantalla –como el de la imagen 21– informando de los permisos que debe aceptar, con las opciones de aceptarlas e ir a la aplicación, o cancelar y volver a la pantalla principal de Facebook.



Imagen 21. Confirmación de permisos de la aplicación

Una vez que se han comprobado que los permisos hayan sido aceptados, la aplicación carga el SDK de JavaScript e inicializa la aplicación, comprobando que la identificación de la aplicación y otros parámetros son correctos. A continuación, muestra ya la interfaz gráfica, tal y como se muestra en imagen 22.

UC3M

BUSQUEDA Y GRAFO

TFG 11/12
Gº Ing. Informática
José Ángel Cuadrado

Busca la información que necesitas

Nivel de búsqueda: ☒ Amigos ☐ Amigos de amigos

Buscar

Limpiar

Ayuda

Aquí se mostrará el grafo resultante de tu búsqueda.

Aquí se mostrarán los mejores resultados obtenidos durante la búsqueda en forma de fotografía y agrupados por el tipo de resultado.

Imagen 22. Interfaz gráfica inicial de la aplicación

Dentro de la aplicación, existen los diferentes casos que se han ido describiendo en los casos de uso y en los requisitos de usuario y de software. Por un lado, al pulsar el botón de “Ayuda”, se activa una alerta que muestra toda la sección para ayudar al usuario a utilizar la aplicación (imagen 23).

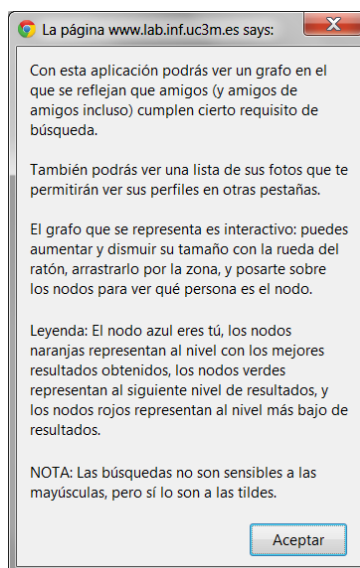


Imagen 23. Sección de ayuda

Una vez que el usuario sabe el funcionamiento de la aplicación, puede comenzar a realizar una búsqueda. Para ello, debe seleccionar primero el nivel de profundidad de la búsqueda: buscar sólo entre sus amigos, o buscar también entre los amigos de sus amigos. Para ello, deberá seleccionar uno de los dos botones de tipo “radio” que hay debajo del cuadro de texto para escoger el nivel de búsqueda.

Finalmente, para realizar la búsqueda, el usuario tendrá que escribir lo que desea buscar entre los datos de sus amigos y amigos de amigos en el cuadro de texto que hay. Si en este cuadro de texto, deja el comentario ya existente, o lo deja vacío o con símbolos de espacio, se muestra un mensaje de alerta indicando que el este texto no es correcto.

Cuando pulse el botón de “Buscar”, empezará a realizar la búsqueda, inhabilitando el botón de “Buscar” y habilitando el botón de “Limpiar”, y mostrando una imagen y un aviso en la zona del grafo que avisa de que el proceso de búsqueda se está ejecutando (reflejado en la imagen 24).

UC3M

BUSQUEDA Y GRAFO

TFG 11/12
Gº Ing. Informática
José Ángel Cuadrado

Busca la información que necesitas

☒ Amigos ☐ Amigos de amigos


Cargando...

Aquí se mostrarán los mejores resultados obtenidos durante la búsqueda en forma de fotografía y agrupados por el tipo de resultado.

Estamos buscando resultados para tu búsqueda según los parámetros especificados. Esta operación tardará entre 1 y 2 minutos en función de los amigos a buscar. (Tiempo máximo de búsqueda: 5 minutos)

Imagen 24. Aplicación en el proceso de búsqueda

6.2. Algoritmo de búsqueda

Una vez que el usuario ha decidido realizar la búsqueda, se empieza a ejecutar un algoritmo que encuentra todas las personas que tienen alguna similitud en los diferentes campos posibles con el texto a consultar por el usuario. Este algoritmo se puede visualizar en la imagen 25 (vista apaisada).

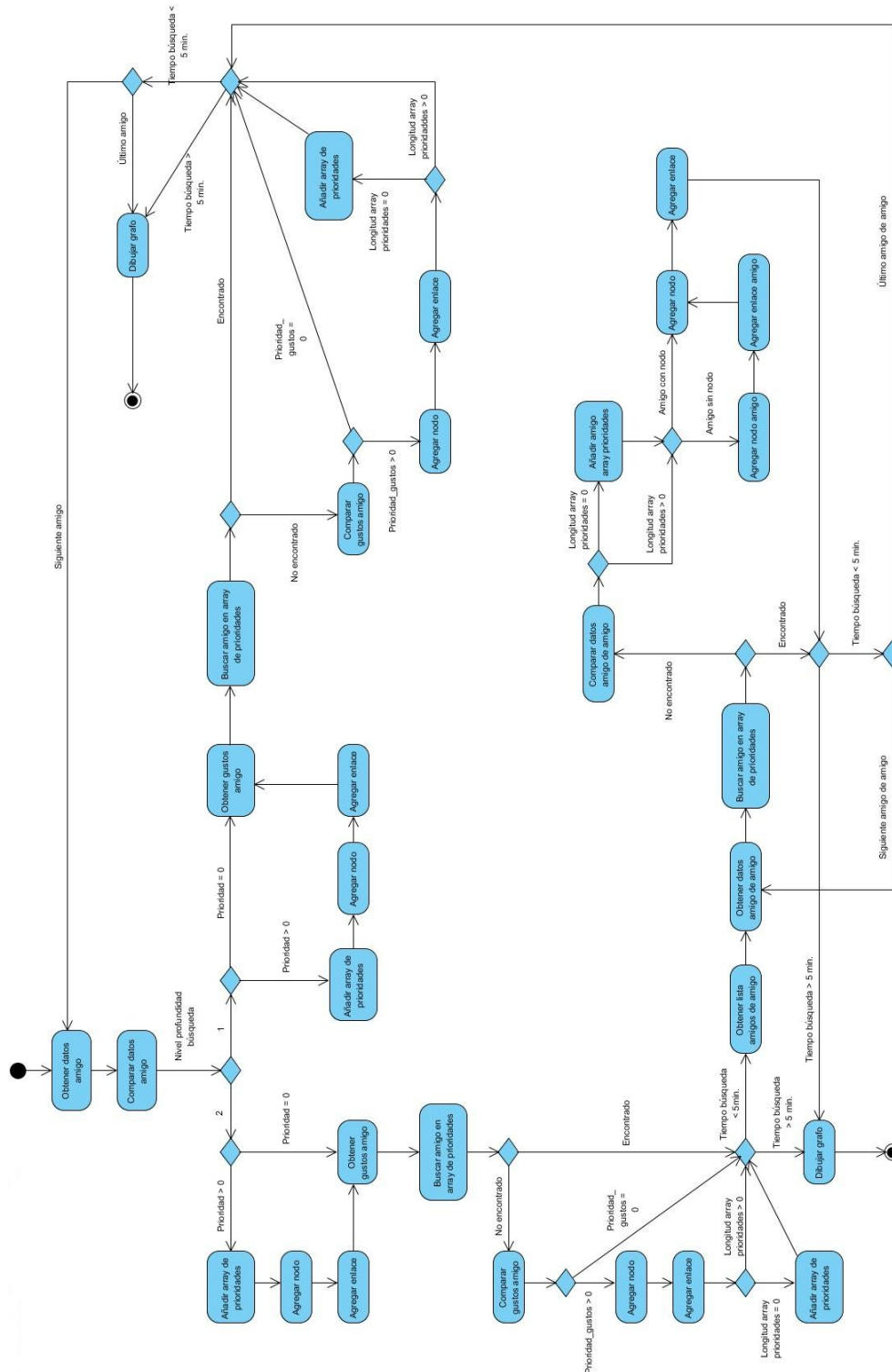


Imagen 25. Algoritmo de búsqueda (vista apaisada)

Antes de explicar el algoritmo, cabe destacar que las llamadas a la API de Facebook son asíncronas, es decir, las funciones y solicitud de datos que se hacen a Facebook se realizan de forma paralela al resto del código. En este caso, tal y como se ha estructurado el algoritmo, se han anidado las llamadas a Facebook, de tal forma que el orden de ejecución son todas las llamadas a los amigos del usuario, a continuación, todas las llamadas a los gustos de los amigos del usuario, y por último, y si se ha solicitado, todas las llamadas a los amigos de los amigos del usuario – primero todos los amigos del primer amigo del usuario, después, todos los amigos del segundo amigo del usuario, y así sucesivamente-.

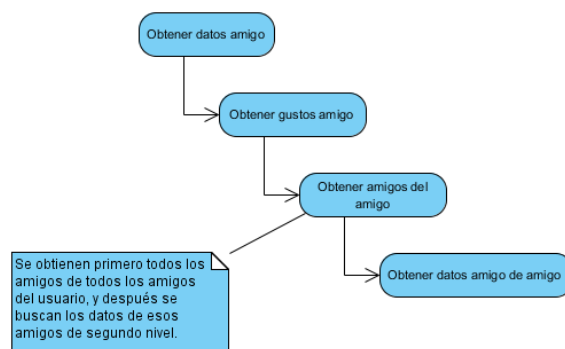


Imagen 26. Llamadas asíncronas de Facebook

El algoritmo empieza obteniendo la información de un amigo. Los datos que se obtienen son comparados con la información que desea encontrar el usuario. Entre los datos con los que se comparará la información en primer grado serán el nombre de la persona, su lugar de nacimiento y de residencia, sus preferencias políticas y religiosas, su estado civil, su historial de trabajo y su historial educativo.

Para realizar esta comparación, lo que se realiza es mirar si el dato introducido por el usuario está contenido en los diferentes apartados (nombre, lugar, clase, descripción, etc.) de los campos mencionados anteriormente.

Sin embargo, el campo de estado civil es especial, ya que este campo está determinado por una serie de palabras clave que ha puesto Facebook. Como estas palabras están en inglés, y la aplicación está basada en el lenguaje castellano, se ha creado un pequeño código, simulando una búsqueda semántica, en la que relaciona distintas posibilidades en castellano con los términos usados por Facebook. De igual forma, al estar los términos en inglés, éstos también se han incluido en el código. En la tabla 69 se puede ver el desarrollo de este código.

Búsqueda en el campo “estado civil”

```

var relacion = info.relationship_status.toLowerCase();
var posRelacion = -1;
switch (str){
    case "soltero":
    case "soltera":

```

```
case "single":
    posRelacion = relacion.indexOf("single");
    break;
case "relación":
case "relationship":
    posRelacion = relacion.indexOf("in a relationship");
    break;
case "comprometido":
case "comprometida":
case "engaged":
    posRelacion = relacion.indexOf("engaged");
    break;
case "casado":
case "casada":
case "married":
    posRelacion = relacion.indexOf("married");
    break;
case "complicado":
case "complicated":
    posRelacion = relacion.indexOf("complicated");
    break;
case "abierto":
case "abierta":
case "open":
    posRelacion = relacion.indexOf("open relationship");
    break;
case "viudo":
case "viuda":
case "widowed":
    posRelacion = relacion.indexOf("open relationship");
```

```
        break;
    case "separado":
    case "separada":
    case "separated":
        posRelacion = relacion.indexOf("separated");
        break;
    case "divorciado":
    case "divorciada":
    case "divorced":
        posRelacion = relacion.indexOf("divorced");
        break;
    case "civil":
    case "union":
        posRelacion = relacion.indexOf("in a civil union");
        break;
    case "domestic":
    case "partnership":
    case "pareja":
    case "hecho":
        posRelacion = relacion.indexOf("in a domestic partnership");
        break;
    }
    if (posRelacion >= 0){
        prioridad = 3;
    }
```

Tabla 69. Búsqueda en el campo "estado civil".

Una vez que se ha hecho la comparación, se comprueba que opción ha escogido el usuario en la profundidad de la búsqueda.

6.2.1. Primer nivel de profundidad

En caso de que el usuario sólo haya seleccionado buscar la información entre sus amigos, la aplicación solamente comprueba los datos del amigo y sus gustos en caso de que no haya encontrado ninguna similitud en los datos anteriores.

Lo primero que realizará el algoritmo es comprobar si se ha encontrado una similitud en los datos anteriores (su prioridad es mayor que 0). En caso de que así sea, añade el identificador de usuario y la prioridad obtenida a un array de prioridades, que servirá después para obtener la lista de mejores resultados. Una vez hecho esto, añade también un nodo y un enlace al objeto JSON creado para recoger los resultados que se mostrarán en el grafo.

A continuación, se obtienen los gustos de ese amigo. Sin embargo, si ese amigo ya ha sido añadido al array de prioridades, no buscará entre los diferentes datos de gustos que se han obtenido (la forma de buscar es igual que con los otros datos de los usuarios). En caso de que no estuviera añadido, se busca entre esos gustos, y si obtiene alguna similitud, lo añade al array de prioridades (siempre y cuando no haya nadie añadido en este array) y añade también el nodo y el enlace al objeto JSON. Si este amigo era el último, se pasa directamente al proceso de dibujar el grafo; en caso contrario, antes de continuar, se comprueba si han pasado 5 minutos desde que se inició la búsqueda. En caso de que así sea, se llama al método para dibujar el grafo.

6.2.2. Segundo nivel de profundidad

Este nivel de profundidad tiene el mismo mecanismo hasta el final del proceso, en el que en vez de comprobar si este amigo era el último –si se tiene en consideración el límite del tiempo-, se pasa a buscar en la lista de amigos de ese usuario.

Si no han pasado los 5 minutos, el siguiente paso es obtener la lista de amigos del amigo en cuestión. Por cada amigo de esa lista, se comprueba primero si ya han sido analizados sus datos. Si no se han analizado, se comparan los datos al igual que se hizo con el amigo, comprobando si el dato solicitado por el usuario está contenido entre los diferentes campos descritos anteriormente.

Una vez pasada esta búsqueda, se observa si la prioridad es mayor de 0 (se ha encontrado una similitud). En caso de que así sea, se añade al array de prioridades si éste está vacío. Después, miramos si el amigo está añadido en el objeto JSON de resultados; si no está agregado, se añade el nodo y el enlace del amigo a este objeto. A continuación, añadimos también el nodo y el enlace del amigo del amigo al objeto.

Finalmente, si éste era el último amigo de amigo, y además, es el último amigo, se llama al proceso de dibujar el grafo. En caso contrario, se continúa, excepto si se ha pasado el tiempo máximo de 5 minutos de búsqueda, que también se llama al proceso de dibujar el grafo.

6.3. Muestra de resultados

Una vez que se ha realizado la búsqueda, el siguiente paso es mostrar los resultados. Para ello, una vez que se llama al método dibujar, se comprueba la cantidad de nodos que hay en el objeto JSON resultante. Si hay más de un nodo (es decir, si hay algún resultado aparte del propio usuario que hace la búsqueda), se procede a dibujar el grafo.

Para empezar este proceso, se inicializa el tamaño de ancho y alto de los límites del grafo, además de la paleta de colores que se utilizarán para colorear los nodos del grafo. A continuación, se crea el panel que contendrá el grafo, definiendo la altura, el ancho, desde que punto de altura empieza el panel, etc. Después, se crea una capa dentro del panel en las que se incluyen los nodos y enlaces del objeto JSON de resultados obtenidos, para finalmente, dibujar las líneas y puntos –configurando su tamaño, color de relleno, nombre, etc.- sobre los enlaces y nodos anteriores.

En la imagen 27 se muestra el resultado de una búsqueda con un segundo nivel de profundidad en la que salen los tres tipos de nodos posibles.

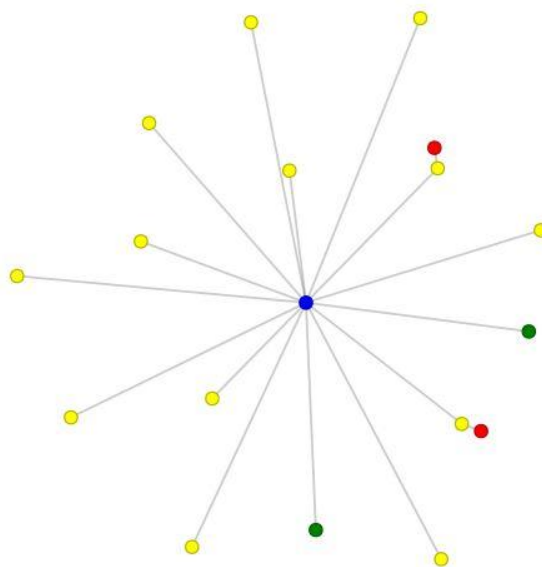


Imagen 27: Grafo resultante de una búsqueda

En caso de que no se haya encontrado ningún resultado en el objeto de resultados, la aplicación muestra en el lugar del grafo un mensaje indicando que “no hay resultados reseñables para esta búsqueda”.

Finalmente, una vez que se ha dibujado el grafo, se pasa a mostrar la lista de mejores resultados a la derecha de éste. Para ello, se llama a otro método que recorre el array de prioridades, y va mostrando de forma agrupada en función de la prioridad que tiene asignada cada persona, las fotos de perfil de cada uno de esos usuarios. En la imagen 28 se muestra el resultado de mostrar esta lista para otra búsqueda realizada en la aplicación.



Imagen 28. Mejores resultados obtenidos de una búsqueda

En la imagen 29, se puede ver un ejemplo del resultado de una búsqueda, incorporando el grafo y la lista de mejores resultados a la interfaz mostrada anteriormente.

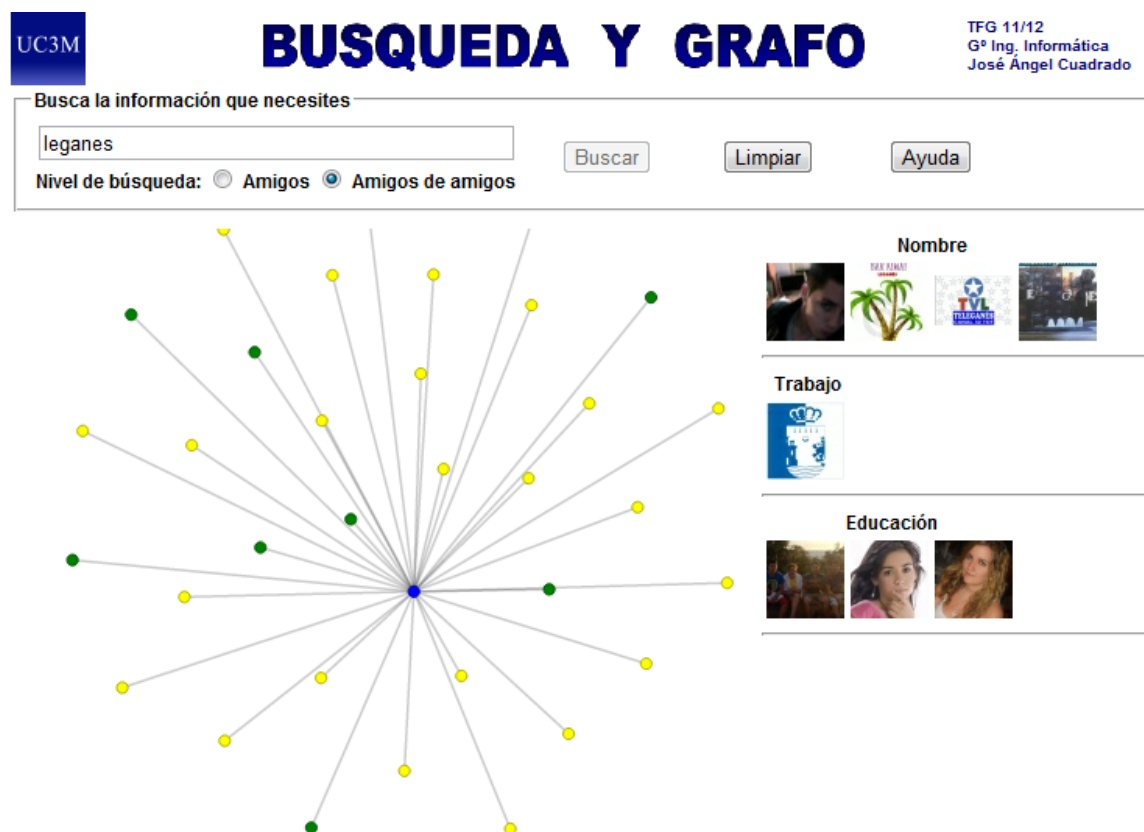


Imagen 29. Resultado final de una búsqueda

7. Validación y verificación del software

El objetivo de este apartado es definir un plan de pruebas que compruebe el correcto funcionamiento del sistema en función de los requisitos solicitados por el cliente.

7.1. Introducción

En esta fase, se identificarán los diferentes elementos que han de ser probados para dar el sistema como aceptado. Para ello, se desarrollará un plan de pruebas que compruebe todas las funcionalidades del sistema. Dichas pruebas serán detalladas, incluyendo los diferentes pasos a seguir que se tienen que dar en las pruebas para que estas se den por aceptadas.

Además del plan de pruebas, se detallarán una serie de encuestas realizadas a usuarios finales para que prueben ellos mismos la aplicación y den su opinión acerca de la utilidad y funcionamiento de la herramienta, para poder realizar mejoras de la aplicación, y obtener líneas futuras para próximas versiones.

7.2. Plan de pruebas

Para asegurar un cierto nivel de calidad y un funcionamiento correcto de la aplicación, será necesario establecer un plan de pruebas. En este apartado, se definirán las diferentes pruebas a realizar, teniendo en cuenta los diferentes elementos que intervienen en la aplicación, las necesidades del entorno y el criterio de aceptación o rechazo de cada prueba.

7.2.1. Elementos de prueba

Para realizar las diferentes pruebas necesarias para la aceptación de la aplicación, será necesaria probar todos los elementos de ésta. Para el caso que ocupa, el sistema contiene un único componente que está conectado a la base de datos de Facebook.

Para identificar mejor los elementos de prueba, se tomará como referencia los requisitos de capacidad que proporcionó el cliente en el punto “3.4.1. *Requisitos de capacidad*”, y de esta forma, probar todas sus solicitudes y asegurar que el sistema que solicitó funciona correctamente.

Teniendo en cuenta esto, las características que se probarán serán las siguientes:

- Conexión con Facebook
- Conexión con la aplicación
- Conexión segura
- Solicitud de permisos
- Carga de la interfaz gráfica
- Consultar ayuda
- Imagen en proceso de búsqueda
- Búsqueda para nivel 1 de profundidad
- Búsqueda para nivel 2 de profundidad
- Alejar y acercar el grafo
- Mostrar el nombre del nodo del grafo
- Arrastrar el grafo
- Arrastrar el nodo
- Mostrar agrupados los mejores resultados
- Limpiar el grafo

Para realizar todas estas pruebas, se tendrán que realizar una serie de tareas que permitan definirlas para posteriormente ejecutarlas y registrar sus resultados.

A la hora de definir las pruebas, será necesario describir el objetivo de la prueba, en qué estado se encuentra el sistema para ejecutar dicha prueba, y qué pasos son necesarios para definirla.

Posteriormente, para ejecutar las pruebas, se observarán los resultados que nos muestra la aplicación una vez se han seguido los pasos descritos en la prueba. Los resultados que se hayan obtenido serán registrados en una plantilla que se mencionará más adelante.

7.2.2. Necesidades del entorno

Es necesario definir un entorno en el que realizar las pruebas que sea similar al entorno de implantación que se haya especificado. De esta manera, se podrá probar de forma más fiable que las pruebas son pasadas correctamente.

Las pruebas de este proyecto se realizarán en el mismo entorno que se va a implantar, puesto que sólo es necesario un ordenador con conexión a Internet y navegadores con JavaScript, tal y como se especificó en el punto *“5.5.1. Especificación de requisitos de implantación”*.

7.2.3. Criterio de aceptación de las pruebas

Una vez que se hayan ejecutado las pruebas, si éstas cumplen y realizan la función que se pide, se dará la prueba como aceptada y se registrará en una plantilla. En caso contrario, se dará como denegada y se tendrá que revisar la funcionalidad para asegurar el correcto funcionamiento del sistema.

7.3. Procedimientos de prueba

En este apartado, se especificarán las diferentes pruebas que debe pasar el sistema para ser aceptado.

Para ello, todas las especificaciones seguirán una plantilla que se muestra en la tabla 70.

Identificador	Descripción
Precondiciones	
Pasos a seguir	

Tabla 70. Plantilla de especificaciones de pruebas

Los campos que se mencionan en la tabla 70 se definen a continuación:

- **Identificador:** Identifica la prueba. Sigue el formato PR-XX, donde XX es un número ordinal.
- **Descripción:** Especifica el objetivo de la prueba.
- **Precondiciones:** Estado del sistema previo a la realización de la prueba.
- **Pasos a seguir:** Pasos que se deben realizar durante la prueba para realizar su ejecución y probar su funcionamiento.

PR-01	Arrancar la aplicación correctamente desde Facebook
Precondiciones	<ul style="list-style-type: none"> • Estar conectado a Facebook.
Pasos a seguir	<ol style="list-style-type: none"> 1. Insertar la URL de la aplicación que se encuentra en Facebook. 2. Comprobar que la aplicación arranca correctamente.

Tabla 71. PR-01

PR-02	Arrancar la aplicación correctamente desde la página web
Precondiciones	---
Pasos a seguir	<ol style="list-style-type: none"> 1. Insertar la URL de la aplicación que se encuentra en el servidor. 2. Comprobar que la aplicación arranca correctamente.

Tabla 72. PR-02

PR-03	Arrancar la aplicación correctamente con una conexión segura
Precondiciones	<ul style="list-style-type: none"> • Estar conectado a Facebook.
Pasos a seguir	<ol style="list-style-type: none"> 1. Insertar la URL segura de la aplicación que se encuentra en Facebook. 2. Comprobar que la aplicación arranca correctamente.

Tabla 73. PR-03

PR-04 Solicitud de permisos

Precondiciones	<ul style="list-style-type: none"> • Estar conectado a Facebook. • No se han aceptado anteriormente los permisos.
Pasos a seguir	<ol style="list-style-type: none"> 1. Arrancar la aplicación correctamente. 2. Comprobar que la aplicación solicita los permisos de la aplicación.

Tabla 74. PR-04

PR-05 Cargar la interfaz gráfica una vez haya arrancado la aplicación

Precondiciones	<ul style="list-style-type: none"> • Haber aceptado los permisos de la aplicación.
Pasos a seguir	<ol style="list-style-type: none"> 1. Comprobar que la aplicación muestra su interfaz correctamente.

Tabla 75. PR-05

PR-06 Consultar la ayuda de la aplicación

Precondiciones	<ul style="list-style-type: none"> • Haber cargado la interfaz gráfica de la aplicación.
Pasos a seguir	<ol style="list-style-type: none"> 1. Pulsar el botón “Ayuda”. 2. Comprobar que se muestra una alerta en pantalla con la ayuda del sistema.

Tabla 76. PR-06

PR-07 Mostrar una imagen durante el proceso de búsqueda

Precondiciones	<ul style="list-style-type: none"> • Realizar una búsqueda
Pasos a seguir	<ol style="list-style-type: none"> 1. Escribir algún dato en el campo de texto. 2. Seleccionar el nivel de profundidad de la búsqueda. 3. Pulsar el botón “Buscar”. 4. Comprobar que se muestra una imagen durante el proceso de búsqueda que informe sobre la realización de este proceso.

Tabla 77. PR-07

PR-08 Búsqueda para nivel 1 de profundidad

Precondiciones	<ul style="list-style-type: none"> • Realizar una búsqueda
Pasos a seguir	<ol style="list-style-type: none"> 1. Escribir algún dato en el campo de texto. 2. Seleccionar el nivel 1 de profundidad de la búsqueda. 3. Pulsar el botón “Buscar”. 4. Comprobar que se muestran los resultados en los que sólo salen amigos del usuario.

Tabla 78. PR-08

PR-09 Búsqueda para nivel 2 de profundidad

Precondiciones	<ul style="list-style-type: none"> • Realizar una búsqueda que obtenga resultados para amigos de amigos
Pasos a seguir	<ol style="list-style-type: none"> 1. Escribir algún dato en el campo de texto. 2. Seleccionar el nivel 2 de profundidad de la búsqueda. 3. Pulsar el botón “Buscar”. 4. Comprobar que se muestran los resultados en los que salen amigos de amigos del usuario.

Tabla 79. PR-09

PR-10 Alejar y acercar el grafo

Precondiciones	<ul style="list-style-type: none"> Realizar una búsqueda correctamente
Pasos a seguir	<ol style="list-style-type: none"> Utilizar la rueda del ratón hacia atrás/adelante Comprobar que el grafo se aleja/acerca.

Tabla 80. PR-10

PR-11 Mostrar el nombre de un nodo del grafo

Precondiciones	<ul style="list-style-type: none"> Realizar una búsqueda correctamente
Pasos a seguir	<ol style="list-style-type: none"> Posar el cursor del ratón sobre un nodo Comprobar que se muestra el nombre del nodo.

Tabla 81. PR-11

PR-12 Arrastrar el grafo por la zona delimitada

Precondiciones	<ul style="list-style-type: none"> Realizar una búsqueda correctamente
Pasos a seguir	<ol style="list-style-type: none"> Pulsar el ratón sobre una zona del grafo. Mantener pulsado el ratón y arrastrar por la zona. Comprobar que se mueve el grafo siguiendo el movimiento del ratón.

Tabla 82. PR-12

PR-13 Arrastrar un nodo por la zona delimitada

Precondiciones	<ul style="list-style-type: none"> Realizar una búsqueda correctamente
Pasos a seguir	<ol style="list-style-type: none"> Pulsar el ratón sobre un nodo del grafo. Mantener pulsado el ratón y arrastrar por la zona. Comprobar que se mueve el nodo del grafo siguiendo el movimiento del ratón.

Tabla 83. PR-13

PR-14 Mostrar agrupados los mejores resultados obtenidos

Precondiciones	<ul style="list-style-type: none"> Realizar una búsqueda correctamente
Pasos a seguir	<ol style="list-style-type: none"> Comprobar que la lista de mejores resultados está dividida en varias secciones en función de los resultados obtenidos.

Tabla 84. PR-14

PR-15 Limpiar el grafo

Precondiciones	<ul style="list-style-type: none"> Realizar una búsqueda correctamente
Pasos a seguir	<ol style="list-style-type: none"> Pulsar el botón “Limpiar” Comprobar que se refresca la pantalla y el sistema está listo para realizar otra búsqueda.

Tabla 85. PR-15

7.4. Análisis de consistencia de pruebas

En este apartado, se realizará una matriz de trazabilidad entre las pruebas definidas y los requisitos de capacidad definidos anteriormente. De esta forma, se analiza si se han llevado a cabo casos de pruebas para todos los requisitos funcionales solicitados por el cliente.

Para que las pruebas pasen este análisis de consistencia, cada prueba debe tener asignado al menos un requisito de capacidad, y viceversa.

	RU-C-01	RU-C-02	RU-C-03	RU-C-04	RU-C-05	RU-C-06	RU-C-07	RU-C-08	RU-C-09	RU-C-10
PR-01	X									
PR-02	X									
PR-03										X
PR-04	X	X	X							
PR-05		X	X	X		X	X		X	
PR-06									X	
PR-07				X						
PR-08		X	X		X					
PR-09		X	X		X					
PR-10						X		X		
PR-11						X		X		
PR-12						X		X		
PR-13						X		X		
PR-14							X			
PR-15		X								

Tabla 86. Matriz de trazabilidad PR-RC

7.5. Informe de pruebas

En este apartado, se realizarán diferentes ejecuciones de la aplicación para poder seguir los pasos descritos en las diferentes pruebas mencionadas anteriormente.

Para la realización de este informe de pruebas, se llevará a cabo una plantilla en la que describir los resultados de éstas. Esta plantilla es la descrita en la tabla 87.

Identificador	Descripción
Fecha	
Responsable	
Resultado	
Fallos	

Tabla 87. Plantilla de informe de pruebas

Los campos que se mencionan en la tabla 87 se describen a continuación:

- **Identificador:** Identifica el caso de prueba. Sigue el formato R-PR-XX, donde XX es un número ordinal.
- **Descripción:** Especifica el objetivo de la prueba.
- **Fecha:** Fecha en la que se realizó la prueba.
- **Responsable:** Miembro del equipo de proyecto que realizó la prueba.
- **Resultado:** “Aceptado” o “Rechazado”, en función del resultado de la prueba.
- **Fallos:** Descripción de los fallos que hayan podido surgir en caso de que la prueba haya sido rechazada.

R-PR-01

Arrancar la aplicación correctamente desde Facebook

Fecha	31 de Mayo de 2012
Responsable	José Ángel Cuadrado Mingo
Resultado	Aceptado: Se ha introducido la URL de la aplicación desde Facebook (http://apps.facebook.com/busquedaygrafo/) y se ha conectado correctamente a la aplicación.
Fallos	---

Tabla 88. R-PR-01



Imagen 30. Resultado de la prueba 1

R-PR-02

Arrancar la aplicación correctamente desde la página web

Fecha	31 de Mayo de 2012
Responsable	José Ángel Cuadrado Mingo
Resultado	Aceptado: Se ha introducido la URL de la aplicación desde la web (http://www.lab.inf.uc3m.es/~a0080563/busquedaygrafo/) y se ha conectado correctamente a la aplicación.
Fallos	---

Tabla 89. R-PR-02



Imagen 31. Resultado de la prueba 2

R-PR-03

Arrancar la aplicación correctamente desde una conexión segura

Fecha

4 de Junio de 2012

Responsable

José Ángel Cuadrado Mingo

Resultado

Aceptado: Se ha introducido la URL de la aplicación desde Facebook con una conexión segura (<https://apps.facebook.com/busquedaygrafo/>) y se ha conectado correctamente a la aplicación.

Fallos

Tabla 90. R-PR-03



Imagen 32. Resultado de la prueba 3

R-PR-04

Solicitud de permisos

Fecha

6 de Junio de 2012

Responsable

José Ángel Cuadrado Mingo

Resultado

Aceptado: Se ha conectado con la aplicación sin haber aceptado los permisos, y al intentar entrar, se solicita la aceptación de éstos.

Fallos

Tabla 91. R-PR-04



Imagen 33. Resultado de la prueba 4

R-PR-05	Cargar la interfaz gráfica una vez haya arrancado la aplicación
Fecha	31 de Mayo de 2012
Responsable	José Ángel Cuadrado Mingo
Resultado	Aceptado: Una vez aceptados los permisos, al entrar en la aplicación, se carga la interfaz gráfica.
Fallos	---

Tabla 92. R-PR-05



Imagen 34. Resultado de la prueba 5

R-PR-06	Consultar la ayuda de la aplicación
Fecha	31 de Mayo de 2012
Responsable	José Ángel Cuadrado Mingo
Resultado	Aceptado: Cuando se pulsa el botón “Ayuda”, se muestra una alerta con la ayuda del sistema.
Fallos	---

Tabla 93. R-PR-06



Imagen 35. Resultado de la prueba 6

R-PR-07**Mostrar una imagen durante el proceso de búsqueda**

Fecha

4 de Junio de 2012

Responsable

José Ángel Cuadrado Mingo

Resultado

Aceptado: Cuando se realiza una búsqueda, durante el proceso de ésta se muestra una imagen y un aviso en pantalla indicando que se está realizando la búsqueda.

Fallos

Tabla 94. R-PR-07



Imagen 36. Resultado de la prueba 7

R-PR-08**Búsqueda para nivel 1 de profundidad**

Fecha

4 de Junio de 2012

Responsable

José Ángel Cuadrado Mingo

Resultado

Aceptado: Cuando se realiza una búsqueda con nivel 1 de profundidad, muestra los resultados buscando entre los datos de los amigos.

Fallos

Tabla 95. R-PR-08



Imagen 37. Resultado para la prueba 8

R-PR-09

Búsqueda para nivel 2 de profundidad

Fecha

6 de Junio de 2012

Responsable

José Ángel Cuadrado Mingo

Resultado

Aceptado: Cuando se realiza una búsqueda con nivel 2 de profundidad, muestra los resultados buscando entre los datos de los amigos y los amigos de amigos.

Fallos

Tabla 96. R-PR-09

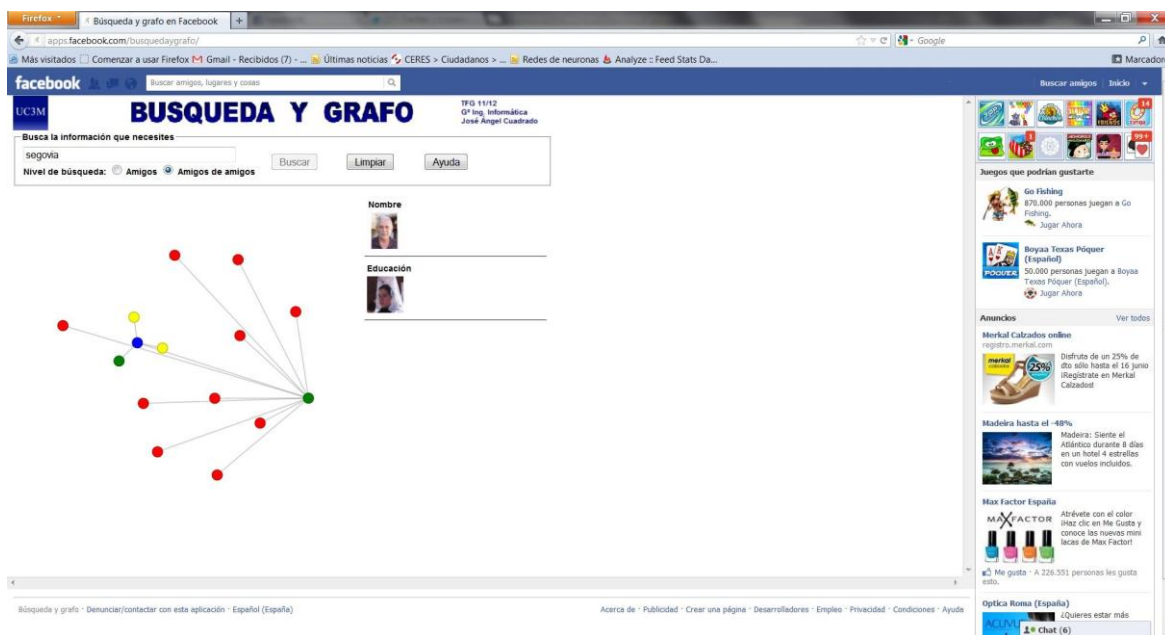


Imagen 38. Resultado para la prueba 9

R-PR-10

Alejar y acercar el grafo

Fecha

5 de Junio de 2012

Responsable

José Ángel Cuadrado Mingo

Resultado

Aceptado: Cuando se muestra el grafo, éste puede alejarse y acercarse con la rueda del ratón

Fallos

Tabla 97. R-PR-10



Imagen 39. Resultado para la prueba 10 (cerca)



Imagen 40. Resultado para la prueba 10 (lejos)

R-PR-11

Mostrar el nombre de un nodo del grafo

Fecha

4 de Junio de 2012

Responsable

José Ángel Cuadrado Mingo

Resultado

Aceptado: Cuando se muestra el grafo, se posa el cursor sobre un nodo para que aparezca el nombre de éste.

Fallos

Tabla 98. R-PR-11



Imagen 41. Resultado para la prueba 11

R-PR-12

Arrastrar el grafo por la zona delimitada

Fecha

4 de Junio de 2012

Responsable

José Ángel Cuadrado Mingo

Resultado

Aceptado: Cuando se muestra el grafo, se arrastra éste a otra zona.

Fallos

Tabla 99. R-PR-12



Imagen 42. Resultado para la prueba 12 (zona original)



Imagen 43. Resultado para la prueba 12 (zona nueva)

R-PR-13

Arrastrar un nodo por la zona delimitada

Fecha

4 de Junio de 2012

Responsable

José Ángel Cuadrado Mingo

Resultado

Aceptado: Cuando se muestra el grafo, se arrastra un nodo de éste a otra zona.

Fallos

Tabla 100. R-PR-13



Imagen 44. Resultado para la prueba 13 (nodos originales)



Imagen 45. Resultado para la prueba 13 (nodo arrastrándose)

R-PR-14

Mostrar agrupados los mejores resultados obtenidos

Fecha

5 de Junio de 2012

Responsable

José Ángel Cuadrado Mingo

Resultado

Aceptado: Cuando se realiza la búsqueda, los mejores resultados obtenidos aparecen agrupados por el motivo de su similitud.

Fallos

Tabla 101. R-PR-14



Imagen 46. Resultado para la prueba 14

R-PR-15**Limpiar el grafo**

Fecha	5 de Junio de 2012
Responsable	José Ángel Cuadrado Mingo
Resultado	Aceptado: Una vez realizada la búsqueda, para realizar otra, se limpian los resultados de la pantalla.
Fallos	---

Tabla 102. R-PR-15



Imagen 47. Resultado para la prueba 15 (sin limpiar)



Imagen 48. Resultado para la prueba 15 (limpiado)

7.6. Pruebas de usuario

Para terminar de realizar las pruebas sobre la aplicación, ésta debe de pasar una serie de pruebas con usuarios finales de verdad. Para ello, se ha realizado una encuesta pública a la que ha sido invitada gente de toda edad, ocupación profesional y experiencia en las redes sociales.

Para poder realizar estas pruebas, se ha pedido a estas personas que prueben primero la aplicación, y que a continuación contesten a un sencillo cuestionario. En este cuestionario, se hacen preguntas sobre diferentes aspectos, como su opinión acerca de la aplicación y su utilidad, si la usarían y se la recomendarían a sus amigos, qué es lo que más le gusta y lo que menos, y que cosas mejorarían en ella.

De todas las personas invitadas, han contestado a la encuesta un total de 25. En los gráficos encontrados en el “*Anexo C*”, se podrá ver el tipo de personas que ha participado en esta encuesta.

Según estos datos, hay un reparto equitativo entre personas de diferentes sexos (60% masculino frente al 40% femenino, según la imagen 50), existe una clara tendencia del público joven (84% de las personas están en un rango de edad de entre 20 y 27 años, según la imagen 51), y la gran mayoría de los encuestados son ahora mismo estudiantes universitarios, entre los que destaca la ingeniería como campo de estudio (60% de estudiantes de ingeniería, 24% de estudiantes universitarios de otras ramas, y 16% de personas trabajando, según la imagen 52).

Los datos referentes a la edad reflejan el hecho de que las redes sociales son usadas en su gran mayoría por gente de una edad joven, mientras que el hecho de que tengamos muchos estudiantes universitarios de un ámbito ingenieril, nos permite analizar la herramienta de una forma más técnica.

En cuanto a las preguntas de la encuesta en sí, en primer lugar se pregunta acerca de la opinión general de la aplicación y de si el usuario le encuentra una utilidad. La gran mayoría de las personas contesta que sí que le encuentra utilidad, ya que se puede buscar a personas sobre algo concreto que desees buscar de una forma rápida y cómoda. Además, ha gustado también la posibilidad de encontrar información sobre amigos de amigos, y opinan que esta aplicación también podría ser usada por las empresas para filtrar en la base de datos de Facebook para, por ejemplo, realizar publicidad en Internet a cierto sector de personas.

Al hilo de esta pregunta, en la encuesta también se pregunta si utilizaría esta herramienta para encontrar a personas relacionadas con un tema o si se lo recomendaría a un amigo. En este caso, también se obtiene una respuesta bastante positiva, puesto que una mayoría de los usuarios que han probado la aplicación volverían a utilizarla para buscar a gente y se lo recomendarían a sus amigos. Cabe destacar también que entre los casos en los que la gente da su vista negativa a esta pregunta, algunas personas acaban diciendo que si se mejorara algún aspecto de la aplicación, sí que la utilizarían y/o recomendarían.

Finalmente, se les pregunta a los usuarios qué es lo que más les gusta de la aplicación y lo que menos les gusta, y cómo la mejorarían. En este apartado se ha encontrado variedad de opiniones, llegando a encontrar incluso opiniones contrariadas.

Entre los aspectos positivos de la aplicación se encuentran la lista de mejores resultados agrupada por sus diferentes características y la aparición de las fotos en la lista –de forma que sea más distinguible cada persona-, la posibilidad de encontrar información de amigos de amigos, su facilidad de uso, la interactividad del grafo, incluyendo la distinción de la prioridad en colores, y sobre todo, la idea en sí de la aplicación.

Sin embargo, a pesar de que haya gente a la que le haya gustado la interactividad del grafo y la facilidad de uso de la aplicación, hay otro sector que no opina lo mismo sobre ello, ya que su opinión es que hay bastantes elementos de la interfaz gráfica y de la usabilidad de la aplicación que podrían mejorarse, tales como la posibilidad de que no hubiera que limpiar los resultados para realizar otra búsqueda o que la interactividad del grafo fuera mejor. Esto último puede ser debido a una falta de información por parte del usuario a la hora de utilizar la aplicación, aspecto que también ha sido criticado, bien por no haberlo leído y/o encontrado en la aplicación, o bien por ser insuficiente la información encontrada. Aparte de estos aspectos, también se ha comentado que la aplicación tarda demasiado tiempo en realizar la búsqueda y la cantidad excesiva de permisos que se solicitan.

En cuanto a las posibles mejoras de la aplicación, la gran mayoría de ellas se tratan de temas de usabilidad y de diseño de la interfaz gráfica que de la idea o la funcionalidad de la aplicación en sí. En primero lugar, se solicita que se especifique todo de forma más clara, de manera que quede más claro cómo utilizar la herramienta, así como interpretar los datos obtenidos a través del grafo y la lista de resultados. Otros aspectos de usabilidad a mejorar son la eliminación del botón de “Limpiar” y de borrar el ejemplo que se encuentra en el cuadro de texto cuando se pincha en él. En cuanto al diseño, se ha comentado también que se mejorara la maquetación en general de la aplicación, o que se colocaran fotos en vez de nodos en el grafo. También se pide que se explique cómo funciona el algoritmo de búsqueda y el tiempo de proceso, así como mejorar la funcionalidad del grafo. Por último, también se han incluido comentarios acerca de la posibilidad de invitar a personas a utilizar la aplicación, publicar resultados, etc.

Conociendo estos datos, se pueden mejorar algunos aspectos antes de la exposición final de la aplicación al cliente, tales como una aclaración de la ayuda y una leyenda más visible, pero otros tendrán que quedarse finalmente como líneas futuras de la aplicación.

8. Conclusiones

Cuando se inició este proyecto, se pretendía cumplir una serie de objetivos marcados al principio del mismo, y que poco a poco se han ido desarrollando hasta cumplirlos de forma satisfactoria.

Primero, se consiguió cumplir con el objetivo de crear un programa de búsqueda de relaciones en redes sociales, en concreto, en la red social Facebook, siendo ésta la que más facilidades ofrecía para desarrollar aplicaciones y acceder a su base de datos.

Además, se llevó a cabo el proyecto siguiendo una adaptación de la metodología “Métrica 3”, que permitió realizar el proyecto, tanto en su vertiente de la aplicación como de documentación, con un cierto nivel de calidad.

También se ha seguido una planificación que permitiera cumplir con los plazos de entrega establecidos. Además, se ha tenido en cuenta un marco económico y legal.

Se han utilizado los conocimientos aprendidos durante la carrera, en especial, aquellos dedicados a la gestión de proyectos, redes, programación e interfaces de usuario. Además, se han aprendido nuevos conocimientos, tales como la programación en redes sociales, o mejorado los existentes acerca de estas materias, como el desarrollo de software en JavaScript.

Finalmente, se ha podido comprobar la complejidad de desarrollar un proyecto software completo con todos los elementos que esto requiere, tanto en su elaboración como en el estudio y mantenimiento de éste.

8.1. Líneas futuras

Una vez terminado el desarrollo de la aplicación en sí, se pueden dejar planteadas varias mejoras para una futura versión de la aplicación. Entre todas las ideas, la principal es la de mejorar el algoritmo de búsqueda, de tal forma que ésta sea más eficiente en los resultados encontrados.

Algunas ideas posibles para mejorar el algoritmo es buscar en más campos de la base de datos, diferenciar las palabras que puede introducir un usuario en el campo de texto y no tratarlo como una única cadena, mejorar la forma de calcular la similitud a través de la distancia de Hamming o de modelos de recuperación de información (modelo vectorial, probabilístico, etc.), e incluso, llegar a poder buscar contenidos de forma semántica. También sería importante mejorar el tiempo de respuesta de la búsqueda.

Aparte de esto, otro aspecto a mejorar, tal y como han dicho varios usuarios al realizar la encuesta descrita en apartados anteriores, es la interfaz gráfica. Dentro de ésta, se podría mejorar aspectos del grafo, como poner las fotos en vez de los nodos, poder representar los resultados en varios grafos, o mejorar la interactividad con el grafo.

También se podrían mejorar otros aspectos de la maqueta de la aplicación y su diseño, permitiendo una mayor usabilidad al usuario. Dentro de este aspecto, cabe destacar entre otros la eliminación del botón para limpiar el grafo.

Finalmente, otra opción sería la posibilidad de desarrollar esta aplicación para dispositivos móviles, para que de esta manera, se pudiera buscar a la gente en cualquier momento de necesidad.

8.2. Consideraciones éticas

Cabe destacar que esta herramienta obtiene ciertas informaciones de los usuarios que sean amigos o amigos de amigos de otro usuario. Entre estos datos que maneja la aplicación, existen algunos que son de carácter íntimo, tales como las preferencias religiosas y políticas.

Es por ello que se tiene que tener en cuenta que esta herramienta puede ser utilizada para realizar malas acciones, pudiendo encontrar y clasificar a gente de unos ciertos gustos y/o preferencias.

No obstante, hay que decir también que esto depende de la privacidad que tengan los usuarios puesta en su red social, así como el propio hecho de que esta información la están publicando ellos mismos en su perfil, permitiendo que estos datos sean vistos por cualquier persona.

Por tanto, se llama a la reflexión sobre los datos que se publican en las redes sociales, en especial, en los casos en que estos datos son privados o de carácter íntimo.

Glosario

- **Algoritmo:** Conjunto de operaciones que permiten crear una solución a un problema dado.
- **API:** Acrónimo de “Application Programming Interface”, es un conjunto de funciones que un servicio ofrece a otro programa para poder utilizar dicho servicio en él.
- **Array:** Elemento de programación encargado de almacenar varias variables en una sola y de forma continua.
- **Chat:** Comunicación entre dos o más personas a través de algún programa mediante una conexión a Internet.
- **Cookie:** Información que guarda un servidor sobre un usuario en su equipo.
- **Etiquetar:** Acción en la que un usuario de una red social menciona una relación entre un elemento –normalmente, una fotografía- y otro usuario de la red.
- **Grafo:** Elemento gráfico representado mediante puntos y líneas que permite estudiar las relaciones de unos elementos con otros.
- **Hardware:** Conjunto de elementos que integran la parte física de un ordenador.
- **Interfaz:** Conexión física y/o funcional entre dos sistemas diferentes.
- **JSON:** Acrónimo de “JavaScript Object Notation”, es un objeto con un formato específico para el intercambio de datos en JavaScript.
- **Librería:** Conjunto de funciones y/o programas utilizados para desarrollar otros programas.
- **Open-source:** Códigos y programas que son desarrollados y distribuidos de forma gratuita.
- **Perfil:** Espacio personal de un usuario de una red social en la que se pueden localizar diferentes datos de información personal.
- **Plugin:** Programa que añade una funcionalidad extra a otro.
- **RSS:** Acrónimo de “Really Simple Syndication”, es un formato utilizado para difundir la actualización de los contenidos de una web a los usuarios que estén suscritos a ella.
- **Script:** Pequeño programa que permite la realización de tareas simples.

- SDK: Acrónimo de “Software Development Kit”, es un conjunto de herramientas de desarrollo para que el usuario pueda crear aplicaciones en un entorno concreto.
- Software: Conjunto de programas e instrucciones informáticas para realizar tareas en un ordenador.
- Streaming: Distribución de contenido, normalmente multimedia, en la que el usuario puede utilizar el contenido mientras se lo está descargando.
- Target: Destinatario de cierto servicio.
- URL: Acrónimo de “Uniform Resource Locator”, es una cadena de texto utilizada para nombrar y localizar recursos en Internet.
- Web 2.0: Sitios web que permiten la interactividad entre éstos y sus usuarios.
- Wiki: Sitio web que puede ser editado por varios usuarios a través de Internet.

Bibliografía

- 20minutos.es. (2 de Junio de 2012). *Los 17 puestos de trabajo mejor pagados en Facebook*. Recuperado el 2 de Junio de 2012, de <http://www.20minutos.es/noticia/1491807/0/facebook/trabajos/mejor-pagados/>
- Alexa. (4 de Febrero de 2012). *facebook.com*. Recuperado el 4 de Febrero de 2012, de <http://www.alexa.com/siteinfo/facebook.com>
- Arrington, M. (15 de Julio de 2006). *Odeo releases Twtr*. Recuperado el 2 de Febrero de 2012, de techcrunch.com: <http://techcrunch.com/2006/07/15/is-twtr-interesting/>
- EFE. (4 de Noviembre de 2009). *Twitter ya está disponible en español gracias al trabajo de los internautas*. Recuperado el 2 de Febrero de 2012, de elcomercio.pe: <http://elcomercio.pe/tecnologia/364303/noticia-twitter-ya-esta-disponible-espanol-gracias-al-trabajo-internautas>
- Facebook. (2012). *Aplicaciones*. Recuperado el 5 de Marzo de 2012, de <https://developers.facebook.com/apps>
- Facebook. (2012). *Facebook developers*. Recuperado el 4 de Febrero de 2012, de <https://developers.facebook.com/>
- Facebook. (2012). *Graph API Explorer*. Recuperado el 5 de Marzo de 2012, de <http://developers.facebook.com/tools/explorer>
- Facebook. (2012). *JavaScript SDK*. Recuperado el 5 de Marzo de 2012, de <http://developers.facebook.com/docs/reference/javascript/>
- Facebook. (2012). *JavaScript Test Console*. Recuperado el 5 de Marzo de 2012, de <http://developers.facebook.com/tools/console/>
- Facebook. (23 de Septiembre de 2011). *Política de uso de datos*. Recuperado el 30 de Enero de 2012, de <https://www.facebook.com/about/privacy/>
- Miller, C. C. (28 de Junio de 2011). *Google introduces Facebook competitor, emphasizing privacy*. Recuperado el 27 de Enero de 2012, de The New York Times: http://www.nytimes.com/2011/06/29/technology/29google.html?_r=2
- Mújica, J. E. (8 de Marzo de 2010). *Redes sociales: historia, oportunidades y retos*. Recuperado el 27 de Enero de 2012, de ForumLibertas.com: http://www.forumlibertas.com/frontend/forumlibertas/noticia.php?id_noticia=16428
- Notepad++. (2011). *Notepad++ Home*. Recuperado el 5 de Abril de 2012, de <http://notepad-plus-plus.org/>

- Noticias Jurídicas. (1978). *La Constitución Española de 1978*. Recuperado el 29 de Enero de 2012, de http://noticias.juridicas.com/base_datos/Admin/constitucion.t1.html#a18
- Noticias Jurídicas. (1999). *Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal*. Recuperado el 29 de Enero de 2012, de http://noticias.juridicas.com/base_datos/Admin/lo15-1999.html
- Real Academia Española. (2001). *Diccionario de la lengua española: Privacidad*. Recuperado el 29 de Enero de 2012, de <http://www.rae.es/privacidad>
- Ria Novosti. (2011). *The world map of social networks*. Recuperado el 27 de Enero de 2012, de <http://en.rian.ru/infographics/20110228/162792394.html>
- Romero, M. L. (20 de Enero de 2012). *Google+ supera los 90 millones de usuarios*. Recuperado el 27 de Enero de 2012, de [Geekets.com: http://www.geekets.com/2012/01/google-plus-90-millones-usuarios/](http://www.geekets.com/2012/01/google-plus-90-millones-usuarios/)
- Shelly, G. B., & Frydenberg, M. (2010). *Web 2.0: Concepts and Applications*. Shelly Cashman.
- Simón, A. (11 de Abril de 2008). *Tuenti, la competencia de Facebook hecha en España*. Recuperado el 31 de Enero de 2012, de [cincodias.com: http://www.cincodias.com/articulo/empresas/tuenti-competencia-facebook-hecha-espana/20080411cdscdiemp_36](http://www.cincodias.com/articulo/empresas/tuenti-competencia-facebook-hecha-espana/20080411cdscdiemp_36)
- Stanford Visualization Group. (2012). *Data-Driven Documents*. Recuperado el 15 de Mayo de 2012, de <http://d3js.org/>
- Stanford Visualization Group. (28 de Junio de 2011). *Force-Directed Layouts*. Recuperado el 16 de Marzo de 2012, de <http://mbostock.github.com/protovis/ex/force.html>
- Stanford Visualization Group. (28 de Junio de 2011). *Protovis*. Recuperado el 16 de Marzo de 2011, de <http://mbostock.github.com/protovis/>
- Tuenti. (2012). *Condiciones de uso*. Recuperado el 30 de Enero de 2012, de http://www.tuenti.com/#m=Terms&func=view_terms_of_use
- Twitter. (23 de Junio de 2011). *Política de Privacidad de Twitter*. Recuperado el 30 de Enero de 2012, de <http://twitter.com/privacy>
- Twitter. (2012). *Twitter developers*. Recuperado el 2 de Febrero de 2012, de <https://dev.twitter.com/>
- Wikipedia. (31 de Enero de 2012). *Facebook*. Recuperado el 6 de Febrero de 2012, de <http://es.wikipedia.org/wiki/Facebook>

- Wikipedia. (16 de Septiembre de 2011). *Odeo*. Recuperado el 2 de Febrero de 2012, de <http://en.wikipedia.org/wiki/Odeo>
- Wikipedia. (17 de Diciembre de 2011). *Red Social*. Recuperado el 25 de Enero de 2012, de http://es.wikipedia.org/wiki/Red_social
- Wikipedia. (12 de Diciembre de 2011). *Seis grados de separación*. Recuperado el 27 de Enero de 2012, de http://es.wikipedia.org/wiki/Seis_grados_de_separaci%C3%B3n

ANEXOS

- **Anexo A: Planificación** Pág. 133
Diagrama de Gantt con la planificación del proyecto.
- **Anexo B: Presupuesto** Pág. 134
Plantilla con el presupuesto del proyecto.
- **Anexo C: Datos de la encuesta** Pág. 135
Gráficos y tablas con los datos obtenidos de la encuesta.

Anexo A: Planificación

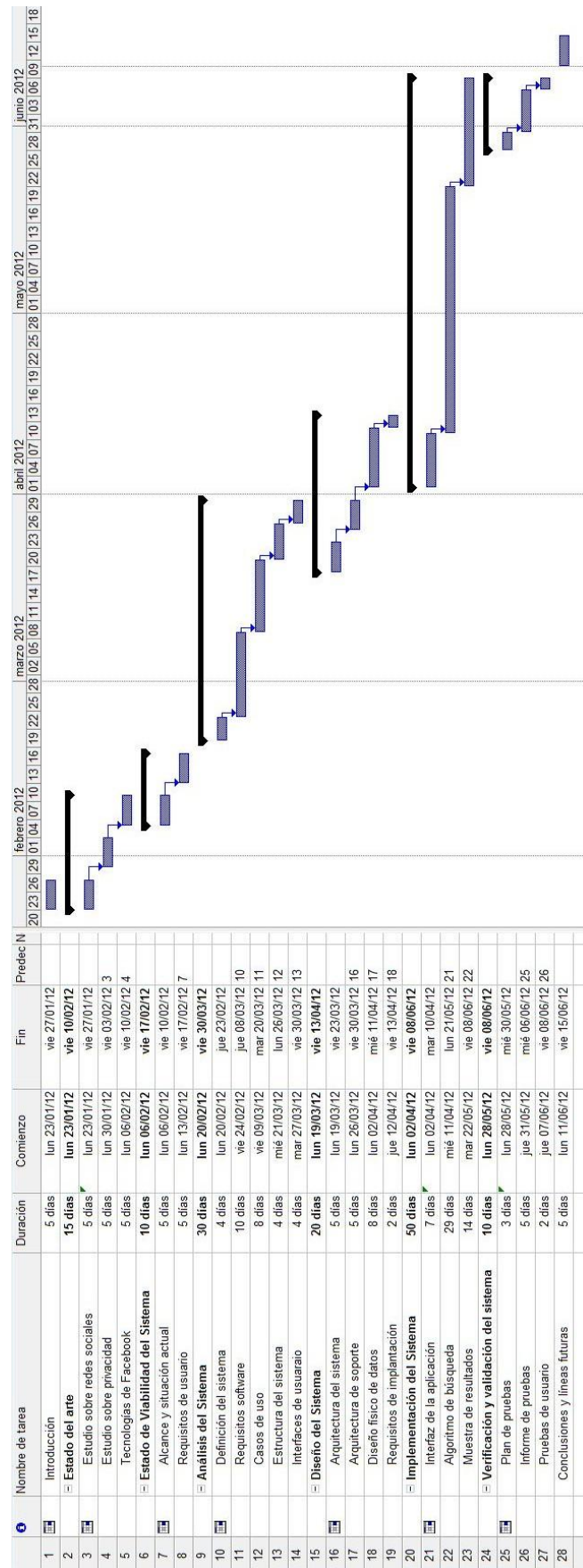


Imagen 49. Planificación (vista apaisada)

Anexo C: Datos de la encuesta

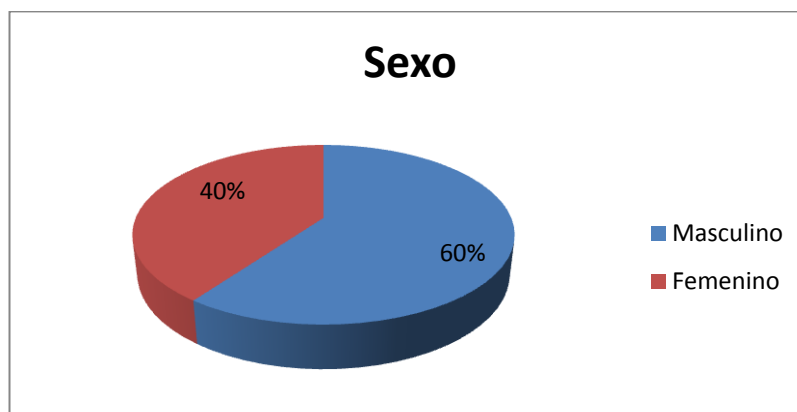


Imagen 51. Porcentaje de “Sexo” en la encuesta

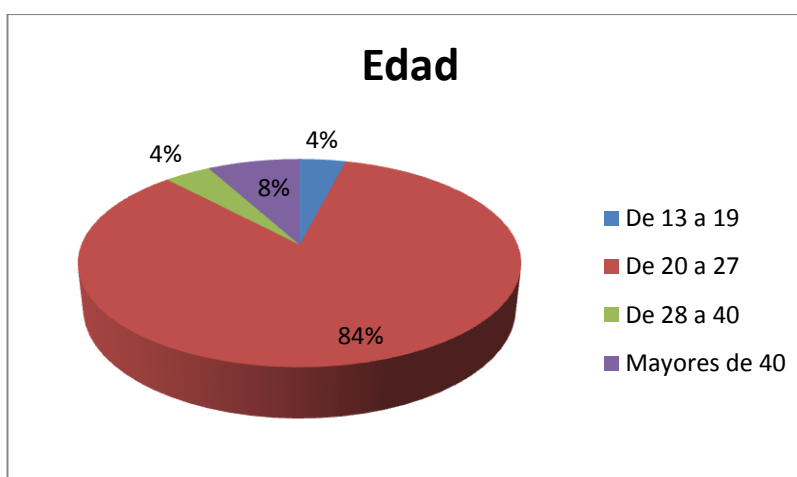


Imagen 52. Porcentaje de “Edad” en la encuesta

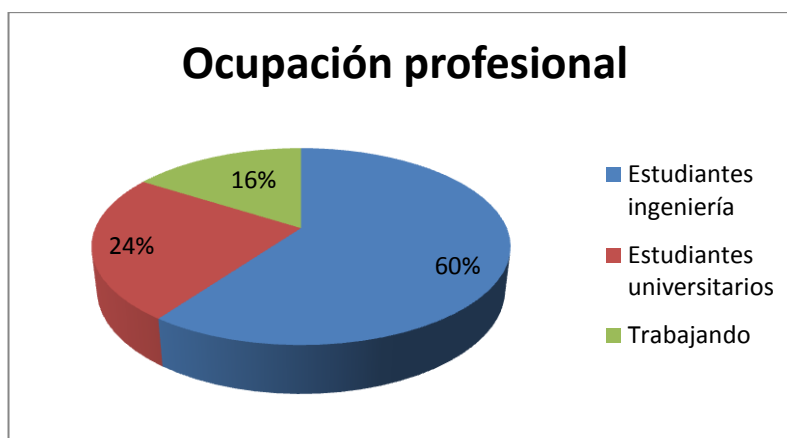


Imagen 53. Porcentaje de “Ocupación profesional” en la encuesta

¿Qué te parece la aplicación? ¿Le encuentras utilidad?
Interesante. Puede ser útil para empresas dedicadas a la publicidad en internet.
Sí, puedo buscar sobre algo concreto y obtener resultados rápidamente, vease una carrera, una serie, un nombre etc.
Es mas fácil buscar así a la gente, no simplemente con el nombre, se parece mas al tuenti en este sentido.
Una aplicación muy útil para buscar relación de datos/intereses
Bastante útil. Sí para encontrar según que tipo de gente.
No me gustan las aplicaciones de las redes sociales, considero que requieren demasiados permisos.
Es una aplicación interesante, que incluye algún aspecto novedoso como el poder buscar entre los amigos de tus amigos y por supuesto, esta función la convierte en una aplicación de bastante utilidad.
Me parece bien, ya que si es útil para encontrar personas con los mismos gustos, profesión, etc.
Si, es un buen filtro de búsqueda
En este contexto, se supone que de las personas que tienes como amigo, conoces sus gustos. De los amigos de amigos, sirve un poco para cotillear. No le encuentro una utilidad real en una red social basada en amigos, aunque si lo que buscas es hacerlos, si que sería de utilidad si las búsquedas las realizara sobre todo facebook.
Le encuentro utilidad a la hora de buscar personas afines a tus gustos.
Me gusta, es entretenida y curiosa, y si tiene utilidad para encontrar los gustos de tus amigos
Me parece original y útil.
si, ya que facebook no tiene un buscador semejante.
SI
Sí bastante, a la hora de buscar gente los filtros siempre ayudan.
Está bien, aunque me surgen algunas dudas. He buscado "uc3m" con amigos de amigos y aunque los resultados del grafo cuadran, en las fotos solo me han salido cuatro resultados, que son los que coinciden con los que tienen "uc3m" como trabajo.
me parece interesante y curiosa
Es interesante y curiosa. Es útil para seleccionar amigos de forma rápida en tu facebook
Me parece útil e interesante.
Parece bastante útil.
Es una manera más fácil de encontrar a las personas que compartan la misma carrera que tú.
Sencilla y fácil de usar
Puede ser util a la hora de filtrar una base de datos de clientes en las empresas.
Curioso, aunque ahora mismo no se le ve un uso específico (demasiado general). Puede ser un motor de búsqueda bueno si se busca por campos (para buscar posibles amigos)

Tabla 103. Pregunta 1 de la encuesta

¿Utilizarías la aplicación si deseas buscar a una persona con cierta característica?
Si la interfaz gráfica cambia los puntos por caras o algun metodo mas rápido de busqueda, si.
Sí
Sí
Sí
Sí
Si, usaría esta aplicación para crear listas de personas con ideas políticas que no me gusten y así poder ejecutarlos con prontitud.
Sin duda sí.
Sí
Si
No en Facebook, pero si por ejemplo en otro tipo de redes sociales, como por ejemplo LinkedIn en la que se buscan profesionales con un cierto perfil
Sí la utilizaría.
Sí, por lo contestado anteriormente
Creo que si la usaría; sobre todo por el gran numero de contactos.
si
Probablemente
Sí, sobretodo si tienes mucha gente los filtros ayudan a encontrarlo más facilmente.
Sí la utilizaría, aunque molaría ver los resultados de formas alternativas y, quizás, más clara.
si
Si
Sí. En las pruebas que he realizado, no siempre encuentra resultados, pero los que encuentra son en su mayoría muy certeros.
Sí.
Sí
Por supuesto
Posiblemente
Con campos probablemente.

Tabla 104. Pregunta 2 de la encuesta

¿Se lo recomendarías a tus amigos?
No lo se.
Sí
Sí
Sí
Sí
Sólo a los que quieran exterminar las mismas etnias que yo.
Sí.
Sí
si
No
La recomendaría, aunque doy por hecho que este tipo de aplicaciones siempre se difunden con cierta facilidad.
Sí.
Si
si
Probablemente
Sí
Sí, aunque salvo los de la Universidad, no creo que ninguno sepa lo que es un grafo xD
si
por supuesto
Si mejorara el tiempo de las búsquedas sí.
Sí.
Sí, porque para ellos también es una herramienta muy útil.
Por supuesto
No, si utilizara esta aplicación sería como parte de mi trabajo de marketing y sería darle pistas a la competencia.
Puede, si se le ve intención de mejora (ahora es muy simple).

Tabla 105. Pregunta 3 de la encuesta

¿Qué es lo que más te gusta de la aplicación? ¿Lo que menos?
El panel derecho de personas relacionadas con la palabra clave esta muy bien. A la aplicación le falta explicació de que es cada cosa.
Lo que más me gusta es la utilidad y la clasificación en gustos, trabajo, educación etc. Buena Funcionalidad. Lo que menos es la interfaz ya que aunque los errores son leves, son incómodos. Me explico: Se podría poner un tooltip en la barra de búsqueda que ya te avisara de que una búsqueda con tildes es diferente a una sin ellas. Para averiguarlo tienes que pinchar en Ayuda y en teoría todos sabemos utilizar un buscador XD por lo que muy poca gente se lo va a leer. Cuando pinchas en la barra del buscador no se elimina el "Por ejemplo: informatica, Carlos III, etc"
Lo que más me gusta es el poder buscar a amigos por características, puesto que es mucho mas fácil. Y lo que menos, que creo que cuando buscas por amigos de amigos solo te busca entre tus amigos.
Lo que más me ha gustado es su eficiencia a la hora de buscar los datos
Lo que más que salen todas las fotos. Lo que menos que no entiendo el gráfico
Saca una lista de personas relacionadas a un campo.
La característica adicional de buscar en amigos de tus amigos.
Lo que más: aplicación útil. Lo que menos: no se que significan los colores (supongo que yo soy el azul, pero no se que significan los demás, tampoco me gusta, o no veo el significado, de que el grafo cada vez se haga más grande (una leyenda estaría bien)
Lo que más: Es fácil de usar y cualquier característica que se te ocurra puede ser válida para encontrar lo que buscas.
Me gusta la idea sobre la que está basada aunque no sea aplicable a este entorno y que los nodos sean de colores en función de la coincidencia con la persona No me gusta la usabilidad: - A la hora de hacer una nueva búsqueda, hay que limpiar, simplemente se podría dar a Enter y que vuelva a hacer la búsqueda con lo que haya escrito. - Que el grafo se mueva queda bonito pero es poco práctico que lo haga porque para ver los nombres, tienes que ir posicionándote sobre los nodos y cada vez se alejan más.
Lo que más me gusta de la aplicación es la facilidad de poder encontrar gente con características o gustos similares a los tuyos con tan sólo una palabra o rasgo. Lo que menos me gusta de la aplicación es que también incluye páginas o perfiles de entidades, no de personas.
Lo que más me gusta de la aplicación es la utilidad que tiene y se ve bien en el grafo. Lo que menos que el grafo se hace grande solo y no me deja ver todos los resultados.
Lo que más me gusta es que se pueda mover el gráfico y sea dinámico. Quizás cambiaría el que el gráfico se acercara poco a poco.
La zona de mejores resultados me gusta y lo oque menos es tener que limpiar la búsqueda con el botón
+gusta: su funcionalidad -gusta: la respuesta tendría que tener más información (etiquetando de forma que en el grafo/foto se vea el "grado de afinidad" con respecto al término buscado...o/y el por qué se ha elegido esa persona por ese término (en dónde se ha encontrado)
La facilidad para encontrar a la gente y lo que menos que no especifica bien qué tipo de característica se puede buscar, por ejemplo, no sé cómo buscar a personas con cierta edad.
Lo que menos: tarda bastante en ejecutar la búsqueda, estaría bien si se pudiera agilizar (ahora mismo no se me ocurre como, pero como siempre, se puede conseguir si renunciamos a una solución óptima).
que sea interactivo el grafo mola
Lo que más me gusta es que el grafo sea interactivo. Lo que menos: La información de los resultados debería ser más visible a simple vista (significado de los colores, etc.)
Lo que más: la visualización en grafo. Lo que menos: los tiempos de búsqueda.
Lo que más es la claridad de toda la información que aparece. Lo que menos, el que para cambiar el tipo de búsqueda por ejemplo de amigos a amigos de amigos haya que hacer una limpieza, y el tiempo que tarda en hacer la búsqueda.

No tarda tanto en encontrar a la gente como dice. Lo que menos, es el diseño, podría ser un poco mas llamativo.
El tiempo de espera :)
Es bastante intuitiva pero poco gráfica, podría ser más bonita pero cumple con su cometido.
La capacidad de relacionarte con los otros usuarios, facilitando la búsqueda de amistades. El movimiento de los nodos (arrastras tu nodo y se choca con otros!!!).

Tabla 106. Pregunta 4 de la encuesta

¿Cómo crees que podría mejorar la aplicación?
Una leyenda de que es cada cosa. Y una interfaz gráfica para el grafo menos abstracta.
1) Se podría poner un tooltip en la barra de búsqueda que ya te avisara de que una búsqueda con tildes es diferente a una sin ellas. Para averiguarlo tienes que pinchar en Ayuda y en teoría todos sabemos utilizar un buscador XD por lo que muy poca gente se lo va a leer. 2) Cuando pinchas en la barra del buscador no se elimina el "Por ejemplo: informatica, Carlos III, etc" y eso puede provocarte errores a la hora de realizar una búsqueda rápida sin "mirar" lo que estás escribiendo ya que por defecto la mayoría de campos input lo borran siempre. 3) Añadido a lo anterior, cuando realizas una búsqueda se desactiva el botón de buscar por lo que si no he obtenido resultados o quiero saber más sobre una misma búsqueda tengo que: - Pulsar en limpiar - Escribir de nuevo la búsqueda (y puedo no acordarme de cómo era exactamente) - Cambiar el rango de búsqueda - Pulsar en buscar Son muchos pasos y se podría reducir únicamente a 1 o 2 si se permitiera cambiar el rango cuando se te muestran los resultados. 4) ¿Por qué el grafo aumenta de tamaño? xD llega un momento que se hace incómodo porque no te deja ver el nombre de algunos nodos, hay que darle a la ruedecita del ratón y para algunas personas puede ser poco intuitivo. 5) Cuando te salen los resultados yo pondría encima un titulito en plan "Resultados para la búsqueda : " , pero vamos, esto ya es menos crítico.
Haciendo que al buscar por amigos de amigos, busque entre los amigos de tus amigos. se podría acortar el tiempo de búsqueda
Más información sobre la búsqueda. Por ejemplo: buscas caca, y que te salga por qué salen esos amigos con caca.
Proporcionando los datos a los que la aplicación tiene acceso, mostrando toda la información personal de tus objetivos.
En mi opinión no veo ningún defecto y me parece una aplicación novedosa, curiosa y sin necesidad de mejora.
A parte de los dicho en el apartado anterior, estaría bien que si ya se ha realizado una búsqueda se pueda realizar una segunda y pulsar el botón limpiar ya que lo primero que hacer es volver a escribir y el botón buscar desaparece deshabilitado Por último, no se en que se basa la búsqueda pero buscando informática pone como amigos relacionados un bar y un taller... BUEN TRABAJO! ÁNIMO YA QUEDA POCO
Ni idea
-Revisar el algoritmo, porque por ejemplo, le pones (sin comillas) "musica" y salen unos resultados, le pones "uc3m" y salen otros resultados. En esas dos búsquedas, hay un resultado que coincide y lo pone como mejor resultado. Ahora bien, siguiendo el ejemplo de búsqueda, pones "musica, uc3m" y no sale ninguno. - Mejorando la usabilidad por ejemplo, dejando el grafo fijo y poniendo una foto pequeña del usuario - Poniendo un pequeño texto explicativo de cómo funciona el algoritmo para que de más confianza sobre la veracidad del resultado. Es decir, en que partes del perfil busca las características que has puesto, en que se basa para decir que un resultado es mejor que otro, y una pequeña leyenda de los colores
Mejoraría la aplicación añadiendo "tags" característicos, o englobando por ciudades, gustos musicales, etc.
Se podría mejorar permitiendo hacer otra búsqueda sin tener que limpiar. Otra mejora que al pinchar en el recuadro para poner el criterio de búsqueda que se borrara automáticamente.
El gráfico se acercara poco a poco; querría que fuese una opción.
Haciendo que al borrar las palabras de búsqueda puedas iniciar una búsqueda nueva sin darle a limpiar.
1) añadir más información (etiquetando de forma que en el grafo/foto se vea el "grado de afinidad" con respecto al término buscado...o/y el por qué se ha elegido esa persona por ese término (en dónde se ha encontrado) 2) una interfaz de usuario con un poquitín más de maquetación (colorido, ubicación de elementos para mejor usabilidad, etc.)
Facilitando cómo hacer cada tipo de búsqueda.
Te planteo varias líneas de mejora: 1. Como ya te comenté, se podría hacer una opción de

búsqueda avanzada para buscar únicamente en determinados campos, lo que además agilizaría la búsqueda. 2. Mejorar los tiempos de búsqueda de otros modos. 3. Mostrar los resultados de formas alternativas, ahora mismo hecho un poco en falta saber por qué hay muchos nodos del grafo que no tienen foto en la derecha. Se podría ver la parte del perfil que coincide con la búsqueda. 4. El grafo podría proporcionar más información. Podría crearse una arista para cada relación entre personas.
con mas informacion para los torpes como yo
Poner las instrucciones de uso más a la vista, pensando en la gente que no utiliza habitualmente estas herramientas.
Mejorar la rapidez, y la interfaz.
Haciendo que el grafo saliera directamente y no que vaya expandiendose.
Más que nada el diseño, como ya he dicho anteriormente.
Poniéndole nombre a los vértices de los grafos según lo que busques
Parseando los Enter para no tener que hacer clic (Evitaría equivoaciones). La haría más grafica y permitiría lo tipico de invitar a amigos a probarla y tal
-Campos de búsqueda. -Permitir buscar de nuevo sin tener que darle a limpiar. -Interfaz de usuario. -App disponible para dispositivos moviles (android, iOS...). -Mostrar información mas detallada de los nodos (solo muestra nombres). -Tildes y otros caracteres. -Leyenda más accesible.

Tabla 107. Pregunta 5 de la encuesta